An e-Lottery Scheme Using Verifiable Random Function*

Sherman S.M. Chow, Lucas C.K. Hui, S.M. Yiu, and K.P. Chow

Department of Computer Science,
University of Hong Kong, Pokfulam, Hong Kong
{smchow, hui, smyiu, chow}@cs.hku.hk

Abstract. A number of e-lottery schemes have been proposed; however, none of them can satisfy all the identified requirements. In particular, some of them require a certain subset of players to remain online or the existence of a trusted third party (TTP) in order to generate the winning number(s) and some suffer from the forgery ticket attack. In this paper, we propose a new e-lottery scheme based on *Verifiable Random Function* that can satisfy all the identified requirements without the presence of TTP, yet the result of this generation is publicly verifiable.

Keywords: e-lottery, verifiable random function, applied cryptography.

1 Introduction

Lottery is a multi-billion dollar industry. Apart from gambling, lottery exists in other forms such as fund-raising coupon for charity. In a typical lottery, there is one dealer and a large number of players. Depending on the rules of the game, players bet on a single number or a combination of numbers chosen from a pre-defined domain. A random process (e.g. drawing of lots) is used to determine the combination of winning numbers. Due to the randomness of the process, it is not repeatable; therefore, the process is usually executed or monitored by a trusted auditing organization. With the popularity of the Internet, it is natural to ask whether we can have a secure e-lottery scheme, that is, having ticket purchase, winning result generation and prize claiming all done over the Internet. While there are many different interpretations of "security" issues in an e-lottery, we believe the following criteria are common to most practical e-lottery schemes.

1. *Random generation of the winning result.*
   Each possible combination of numbers from the domain is equally likely to be the winning result. No player can predict the result better than guessing.

* This research is supported in part by the Areas of Excellence Scheme established under the University Grants Committee of the Hong Kong Special Administrative Region (HKSAR), China (Project No. AoE/E-01/99), grants from the Research Grants Council of the HKSAR, China (Project No. HKU/7144/03E, HKU/7136/04E and HKU/7021/00E), and grants from the Innovation and Technology Commission of the HKSAR, China (Project No. ITS/170/01 and UIM/145).

O. Gervasi et al. (Eds.): ICCSA 2005, LNCS 3482, pp. 651–660, 2005.

2. *The winning result is publicly verifiable.*
   Every player can verify the winning result, i.e. the dealer cannot cheat in the generation process.
3. *The total revenue and the number of winning tickets are publicly verifiable.*
   The amount of winning prizes is usually related to dealer's revenue and hence the number of sold tickets should be publicly verifiable. In some lottery games, the prize for each winning player is not fixed but depends on the total number of winning tickets; hence, the number of winning tickets should also be publicly verifiable.
4. *Forgery of winning ticket is impossible.*
   Either players or the dealer cannot forge a ticket. Players cannot create a winning ticket which they did not purchase. The dealer cannot forge a ticket which is the same as a winning ticket after the winning result is generated.
5. *User is not required to be online for the generation of the winning result.*
   In some previous schemes, the generation of the winning result requires some players to remain online. This is not realistic, especially in a large scale game.
6. *Anonymity of player.*
   Player's anonymity is of paramount importance especially for the winning player(s). If a winning player's anonymity is compromised, he/she may face the threat of blackmail, beggars, etc.
7. *Confidentiality of the ticket's value.*
   In some lottery games, the players bet on a single number or a combination of numbers chosen from a pre-defined domain. In the case more than one winning players bet on the same value, the prize will be shared equally among them. In this setting, we need to keep the confidentiality of the ticket's value, or various attacks are possible. For examples, cutting down a certain ticket value's winning share by intentionally "duplicating" other's ticket value, or eavesdropping the channel between the dealer and the clients to learn the "popularity" of the values and betting on the most "profitable" number.
8. *Fairness on purchase of tickets and claiming of prize.*
   Purchase of tickets and claiming of prizes are based on the principle of fair exchange. That is, either both parties (the dealer and the player in our case) get the other party's item, or no party gets the other party's item at the end of a transaction.
9. *No early registration of player is required.*
   Some schemes (e.g. [6]) require early registration of players to avoid collusion of players. This is an unrealistic requirement since early registration is not required in traditional lottery.

Previous Work: There are many existing e-lottery schemes (e.g. [4, 6, 7, 8, 9, 10, 11, 14, 15]) but only three of them ([6, 11, 15]) address similar issues as our scheme without resort to a trusted third party (TTP)[1]. Some schemes assume a model different from ours: [7] considers the lottery game as the form of a *real-time* interactive game between players and dealer in the casino, while the

[1] TTP may be required for fair exchange.

security of [4] is guaranteed by using *multiple* dealers. Some schemes [8, 9, 14] simply assume the existence of a *trusted random source*, for example, from the random number generation process executed by a TTP. Moreover, [14] does not address the anonymity requirement, the forgery attack and the fairness on the purchase of tickets and claiming of prize; while [9] is shown to be insecure by [8].

The most recent e-lottery proposal is the "electronic national lotteries" [10]; however, a TTP (called "verifier" in their terms) is assumed to take into accounts of all players' winner tickets in the winning result generation (so that the dealer cannot predict the winning result until the last player's decision is made) and will not "insert" any forged tickets to affect the winning result generation. Due to the above mentioned reasons, we mainly compare our scheme with [6, 11, 15]. Table 1 shows a comparison between the existing schemes and our new scheme. None of these three existing e-lottery schemes can satisfy all the requirements that we have identified.

In [6], only the tickets sold in the "critical purchase phase" are used to determine the winning result. This makes forgery of winning ticket possible: After the winning result is generated, the dealer can "insert" the winning ticket to the tickets sold before the critical purchase phase, without altering the winning result. The dealer can do so since the sequence numbers and the timestamps associated with the tickets are not guaranteed to be sequential. For example, when multiple purchase requests are made simultaneously but one of the buyers refuses to pay for the ticket afterwards. Besides, the dealer may simply replace the ticket value of a certain ticket which is under his control and "sold" before the critical purchase phase. These malicious acts may not be noticed easily except there is a player who checks each ticket one by one with previously downloaded tickets list from time to time, which is impractical since the number of tickets sold is potentially in millions. As forgery of winning ticket is possible, the total number of winning tickets can be altered easily.

The first e-lottery scheme that uses delaying function (see the definition in Section 2) to prevent forgery ticket attack is [6]. In fact, our scheme also uses a similar idea to prevent the forgery ticket attack. However, their scheme applies the function only on a portion of the tickets sold while our scheme applies the function to all tickets sold so as to achieve a higher level of security.

Instead of applying delaying function in the winning result generation phase, [11] applies delaying function in the verification phase. Their scheme tries to prevent the forgery ticket attack by imposing a time limit for the prize claiming phase. Players must do the verification through a delaying function immediately afterwards, or they will suffer from losing the game due to the insufficient time for verification. This requirement is not realistic. Based on their scheme, the exact number of winning tickets cannot be accurately calculated unless all the winning tickets are *claimed* accordingly. In addition, this scheme has not addressed the anonymity and fair exchange issues.

The first piece of work integrating fair exchange with e-lottery is [15], this scheme satisfies most of the requirements we have identified. However, their winning result generation requires some players to be online, which make the

Table 1. A Comparison of e-Lottery Schemes

Scheme	Requirement for an e-Lottery Scheme									
	1	2	3	4	5	6	7	8	9	Trusted Third Party *Not* Required?
Verifiable Lotteries [6]	Y	Y	*N*	*N*	Y	*N*	Y[2]	*N*	*N*	Y
Digital Lottery Server [14]	Y	Y	*N*[3]	Y	Y	*N*	*N*	*N*	Y	*N*
Fair e-Lotteries [11]	Y	Y	*N*	Y	*N*	*N*	Y[2]	*N*	Y	Y
Lottery on the Internet [15]	Y	Y	Y	Y	*N*	Y	*N*[3]	Y	Y	Y[1]
Our scheme	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y[1]

scheme not realistic and not robust. For example, the winning result cannot be generated if none of the players participates in the generation process. Also, the winning result may be biased if all participating players collude, which is an important issue for small scale e-lottery game or when the e-lottery game is not yet popular in the beginning. Moreover, they have not considered the security issue of not encrypting the ticket's value.

Our Contribution: In this paper, we propose a new e-lottery scheme which satisfies all of the identified requirements by using the *verifiable random function*. Our scheme makes sure that the lottery dealer generates the winning number(s) based on all the tickets sold. The calculation of the winning number(s) does not depend on the participation of players but it is yet verifiable.

2 Preliminaries

We first discuss the cryptographic primitives used in our scheme.

Verifiable Random Function: *Verifiable Random Function* (hereafter referred as VRF) was introduced by [13]. Basically it is a pseudorandom function [5] providing a non-interactively verifiable proof for the output's correctness.

Based on the notation in [12], a function family $F_{(\cdot)}(\cdot) : \{0,1\}^k \mapsto \{0,1\}^{l(k)}$ is a verifiable random function if there exists polynomial-time algorithms ($\mathtt{G}(\cdot)$, $\mathtt{Eval}(\cdot,\cdot)$, $\mathtt{Prove}(\cdot,\cdot)$, $\mathtt{Verify}(\cdot,\cdot,\cdot,\cdot)$) such that

- $\mathtt{G}(1^k)$ is a probabilistic algorithm which output a secret key SK (the secret seed of the random function) and the corresponding public key PK.
- $\mathtt{Eval}(SK, x)$ is an algorithm that computes the VRF's output $y = F_{PK}(x)$.
- $\mathtt{Prove}(SK, x)$ is an algorithm that computes the proof π that $y = F_{PK}(x)$.
- $\mathtt{Verify}(PK, x, y, \pi)$ is an algorithm that verifies $y = F_{PK}(x)$.

And a VRF has to satisfy the following properties:

- *Uniqueness*: If $\mathtt{Verify}(PK, x, y_1, \pi_1) = \mathtt{Verify}(PK, x, y_2, \pi_2) = 1$, $y_1 = y_2$.
- *Computability*: $\mathtt{Eval}(SK, x) = F_{PK}(x)$ is efficiently computable.

[2] In their settings, players are not betting on any particular value.
[3] But we believe that the scheme may be modified to satisfy this requirement.

- *Provability*: $\texttt{Verify}(PK, x, \texttt{Eval}(SK, x), \texttt{Prove}(SK, x)) = 1$.
- *Pseudorandomness*: The probability that an adversary can tell any bit of $F_{PK}(x)$ for x he chose is negligible even if he has seen the values of many $F_{PK}(x')$ given $x' \neq x$.

Please refer to [3, 12, 13] for more details of VRF, e.g. its construction.

To see how VRF helps in the construction of e-lottery scheme, the following shows some key ideas. The random winning number generation is guaranteed by the pseudorandomness of VRF and the generation result is verifiable by the commitment made by dealer (public key). By keeping the private key secret, players are prevented from cheating even if they have access to the VRF's input.

Delaying Function: *Delaying function* was introduced by [6], which is a function that is moderately hard to compute and cannot be parallelized. The time to compute the function depends on the parameters chosen. It may take several hours or several days to complete, using the fastest existing implementation. The output is *publicly verifiable* by going through the same computation. Please refer to [6] for more details of delaying function, e.g. its construction.

Verifiable Encrypted Signature: *Verifiable Encrypted Signature (VES)* is an encrypted signature which can be proved that the decryption of it gives a certain party's signature on a public message, without revealing the signature. It is called *Certificate of Encrypted Message Being a Signature (CEMBS)* in [1]. VES can be constructed from aggregate signature scheme [2] or ElGamal public key encryption scheme. It is a useful primitive for enabling fair exchange [15].

3 A New e-Lottery Scheme

In this section, we describe our proposed e-lottery scheme. We show how to integrate VRF, hash function and delaying function to build a scheme that satisfies all the identified requirements. To ease our discussion, we assume that we only generate one winning number and each ticket bets on one number. It is easy to extend the scheme to cover a combination of numbers.

3.1 Our Proposed Scheme

Our scheme consists of 5 phases, namely, Setup, Ticket Purchase, Winning Result Generation, Prize Claiming and Player Verification. We assume that the numbers used in the lottery game is $\{1, 2, \ldots, u\}$ and the output domain for the VRF is $\{1, 2, \ldots, v\}$ with $v \geq u$. Let k be the security parameter, $H(\cdot)$ be a cryptographic hash function that maps a bit string of arbitrary length (i.e. $\{0,1\}^*$) into a bit string of fixed length (e.g. 160 bits) and let $H_0(\cdot)$ be another cryptographic hash function that maps $\{0,1\}^k$ to $\{0,1\}^{\lceil \log_2 u \rceil}$.

- Setup:
  1. Dealer generates a secret key SK and a public key PK by the algorithm $\texttt{G}$ with the security parameter k as the input.

2. Dealer publishes the following items.
   (a) Hash functions $H(\cdot)$ and $H_0(\cdot)$, delaying function $D(\cdot)$, and VRF's algorithms: $\mathtt{G}(\cdot)$, $\mathtt{Eval}(\cdot,\cdot)$, $\mathtt{Prove}(\cdot,\cdot)$ and $\mathtt{Verify}(\cdot,\cdot,\cdot,\cdot)$.
   (b) VRF public key PK and dealer's digital certificate.
   (c) The amount of time t in which dealer must release the winning ticket value generated. (This is the input parameter controlling the time complexity of the delaying function.)

– Ticket Purchase:
1. Player chooses his favorite number x and randomly picks a random bit string r from $\{0,1\}^k$. r is kept in secret.
2. Player obtains a sequence number s of the ticket from dealer.
3. Player computes $H_0(r)$ and $H(x||s||r)$, then sends $ticket_i = s||(x \oplus H_0(r))||H(x||s||r)$ to dealer.
4. Dealer publishes every single ticket $ticket_i$.
5. Dealer returns a signed ticket $signed\_ticket_i$ to player to acknowledge the recipient of player's purchase request. Any digital signature scheme which is existentially unforgeable against adaptive chosen message attack can do and the signature does not need to be encrypted.
   (Players do not need to enroll to the public key infrastructure as they need to verify the validity of the signature only.)
6. Dealer links the ticket to a one-way hash chain. This chain could be created by $chain_1 = H(ticket_1)$, $chain_i = H(chain_{i-1}||ticket_i)$ for $i > 1$.
7. Dealer publishes $chain_j$ where j is the number of tickets sold so far.

– Winning Result Generation:
1. Suppose the final value of the hash chain is h, computes $d = D(h)$ by the delaying function, and publishes it.
2. Dealer calculates $(w, \pi) = (\mathtt{Eval}(SK, d), \mathtt{Prove}(SK, d))$.
3. If $w > u$, VRF is applied on $w||d$ with padding bits if necessary (one possible source of bits is VRF public key) again.
4. Dealer publishes (w, π) (and intermediate tuples with number of the times VRF is applied, if VRF is applied more than once) within t units of time after the closing of the lottery session.

– Prize Claiming:
1. If $x = w$, player wins.
2. Player submits (s, r) to dealer (in a secure channel).
3. Dealer checks whether a ticket of $s||(w \oplus H_0(r))||H(w||s||r)$ is received.
4. If so, dealer checks whether the tuple (s, r) has already been published (i.e. the prize has been claimed by someone already).
5. If the prize is not yet claimed, dealer pays the player and publishes r.

– Player Verification:
1. Player checks whether his/her ticket(s) is/are included in the hash chain and checks whether the final output of the hash chain is correct, using the knowledge of $ticket_i$'s.
2. Player verifies the validity of d and whether $w = \mathtt{Eval}(SK, D(h))$ by checking whether $\mathtt{Verify}(PK, w, D(h), \pi) = 1$.
3. Intermediate (w, π) tuples may also be checked if necessary.

4. For each winning ticket published, players verify the validity of $s||(w \oplus H_0(r))||H(w||s||r)$.
5. If any mismatch occurs, player should report to the auditing organization by simply providing the parameters involved in the checking.

3.2 Design Philosophy

The rationales behind some important steps of our scheme are discussed here.

Probability of Winning: Suppose $v = u + b$ is the domain size of the VRF's output. The winning probability ensured by the Step 3 of the winning result generation is $\sum_{i=0}^{\infty}(\frac{b}{u+b})^i \frac{1}{u+b} = \frac{\frac{1}{u+b}}{1-\frac{b}{u+b}} = \frac{1}{u}$. We can control the domain size of VRF's output so the expected time of application of VRF can be kept small.

Mapping of the Winning Result to Winning Ticket: In most of the previous e-lottery schemes, the mapping of the winning result to the ticket value is not discussed in details. There may be a situation that the domain of the generated winning result is not the same as the domain of the ticket value. In particular, if the domain size of the winning result and the domain size of the ticket value are relatively prime, it is not straight-forward how one can ensure the pre-determined winning probability of each possible ticket value. In our scheme, the mapping of the winning result to the winning ticket value is trivial as it is a 1-1 mapping.

Use of VRF Key Pairs: To maximize the security level of the scheme, both dealer and players are prevented from gaining knowledge of some output values of VRF by the generation of a new pair of VRF keys for each lottery session.

Use of Sequence Number: The purpose of using sequence number is not the same as that in [6]. In our scheme, it only serves the purpose of making $H(x||s||r)$ distinct even if two players pick the same x and r. The sequence numbers are not required to be following each other one by one strictly. If the sequence number s is not used, the situation that two different winning players who used the same r may occur (although the probability is not high). In such case, when one of the players claimed the prize, dealer has already known the value of r and hence the dealer can falsely claim that the prize of the second winning player has been already paid. In our current design, even the values x and r associated with two different winning tickets are the same, the corresponding sequence numbers are different and hence the values of $H(x||s||r)$ are different.

Adding Fair Exchange Feature: By using CEMBS/VES, fair exchange between the player and dealer can be ensured. The fair exchange protocol used in [15] can be adopted in our scheme as well. For the purchase of tickets, if the player aborts the transaction, the dealer can send the cipher cash, sequence number s and $signed\_ticket_i$ to the TTP. Then TTP will send the e-cash to the dealer and the $signed\_ticket_i$ to the player. For the prize claiming, if the dealer aborts the transaction, the player can send the cipher cash, $ticket_i$ together with r to the TTP. Then TTP will send the e-cash to the player and r to the dealer.

Since there may be more than one player winning the same prize by betting on the same value x, we note that the e-cash used in the prize claiming should

not be of fixed value. It should be an agreement of the dealer to award the player a certain prize. By the publicly verifiability of the number of winning tickets, the amount of award shared by each player is guaranteed.

Use of Delaying Function: Dealer may cheat by trying to "insert" tickets to the hash chain (and "discard" these newly added tickets if the result is not favorable) until a favorable result is obtained. The use of delaying function prevents this kind of attack. One may argue that if the last ticket is sold much before deadline, then the dealer is able to compute delaying functions twice before deadline. However, it is not so probable in a large scale e-lottery. On the other hand, a malicious dealer can corrupt the late submissions, but this is prevented by the fair exchange protocol[4]. Moreover, we found that all existing e-lottery scheme without TTP employs delaying function ([15] also mentioned the use of delaying function is necessary to make their scheme secure). Schemes that do not employ the delaying function, for instance [10], assumed the integrity of the ticket submission pools is ensured by the TTP.

3.3 Analysis

Assuming the lottery is *pari-mutuel*, i.e. the amount of prize is solely based on the sold tickets, our scheme satisfies all the requirements we have identified.

1. *Random generation of winning number.*
   The random generation of winning number is guaranteed by the VRF's pseudorandomness. Each ticket submission from the players is random, thus nobody can predict the outcome before the time when the last ticket is sold. With the use of delaying function, the dealer cannot bias the generation of winning number by trying to insert tickets into the hash chain one by one. So, the dealer has no way to make sure that which number/player will win or loss even all the numbers purchased by the players are known.
2. *The winning result is publicly verifiable.*
   The winning number generation is verifiable by the provable property of VRF. Besides, the value of hash chain and the output of delaying function are also publicly verifiable.
3. *The total revenue and the number of winning tickets are publicly verifiable.*
   (a) The number of sold tickets is verifiable by the first step of player's verification since each player can check whether his/her ticket(s) is/are included in the hash chain.
   (b) For each ticket sold, the dealer will publish the corresponding number x, so each player can verify the total number of winning tickets easily after the winning result is announced.
4. *Forgery of winning ticket is impossible.*
   (a) The dealer cannot forge winning tickets after the winning number has been generated as all the tickets sold have been published and can be publicly verified by all players.

[4] For denial of service attack by dealer, there are other practices to resolve this issue and it is outside the scope of our discussion.

(b) Players cannot forge tickets that they have not actually purchased. The hash function $H(\cdot)$ makes sure that it is difficult to compute r based on the published ticket $s||(x \oplus H_0(r))||H(x||s||r)$.

5. *User is not required to be online at the time of winning number generation.*
6. *Anonymity of player.*
   Player's identity is not reflected in the process of buying ticket, verification and prize claiming. Players' account number can be encrypted as well.
7. *Confidentiality of the ticket's value.*
   The ticket's value is encrypted. Neither cutting down a certain ticket value's winning share nor eavesdropping for the "global statistics" is possible.
8. *Fairness on purchase of tickets and claiming of prize.*
   Fairness can be achieved by using CEMBS/VES as the protocol used in [15].
9. *No early registration of player is required.*

3.4 Other Issues

There are some other minor issues that have been addressed in the previous e-lottery schemes. We briefly discuss how our scheme addresses these issues.

1. *Total purchase of tickets.*
   It is natural that total purchase of all tickets guarantees a winning probability of 1. This threat can be resolved with external means, e.g. by setting the prize obtained by total purchase is less than the cost of total purchase. Because the total revenue is publicly verifiable, there is no use for the malicious dealer inserting all possible winning values to the hash chain and succeed to claim tie whenever some one claim to have a prize, since it will greatly increase the revenue and hence the value of prize to be given out by the dealer.
2. *Unclaimed tickets.*
   The dealer may not be aware of any unclaimed tickets as the tickets' value are encrypted. However, the player's verification is very efficient (in contrast to the time consuming verification in [11]) and hence it is not a real threat.
3. *Other threats.*
   Bogus server, *system database damage*, *reveal of network address of player*, etc. are not considered as there are other practices to resolve these threats.

4 Conclusion

A new e-lottery scheme, based on the verifiable random function, is proposed. Our scheme satisfies all the identified requirements and does not assume the existence of TTP for winning ticket generation.

To make the e-lottery a reality, the way the e-lottery game is conducted should be similar to a traditional one in which players' interaction with the dealer should be kept as simple and minimal as possible. Our scheme has a high degree of resemblance with a traditional one. Players are neither required to be online at the time of winning ticket generation nor to perform time-intensive operations for checking whether the tickets purchased are indeed winning tickets. Early user

registration is not required as our scheme is secure even in the presence of a large size of colluding players. And players are allowed to buy as many tickets as they want. We hope that the result of this work can help to increase the confidence level of customers in participating in e-lottery games.

It is still an open problem to design a publicly verifiable e-lottery scheme without using delaying function and the online participation of players during the generation of the winning result. Another research direction is to make a delaying function which is efficiently verifiable but the computation is still hard.

References

1. Feng Bao, Robert H. Deng, and Wenbo Mao. Efficient and Practical Fair Exchange Protocols with Off-line TTP. In *IEEE Symposium on Foundations of Security and Privacy*, pages 77–85, 1998.
2. Dan Boneh, Craig Gentry, Ben Lynn, and Hovav Shacham. Aggregate and Verifiably Encrypted Signatures. In *EUROCRYPT 03*, LNCS 2656, pages 416–432. Springer, 2003.
3. Yevgeniy Dodis and Aleksandr Yampolskiy. A Verifiable Random Function With Short Proofs and Keys. In *Public Key Cryptography 05*, LNCS 3386, pages 416–431.
4. Pierre-Alain Fouque, Guillaume Poupard, and Jacques Stern. Sharing Decryption in the Context of Voting or Lotteries. In *Financial Cryptography 00*, LNCS 1962, pages 90–104. Springer, 2001.
5. Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to Construct Random Functions (Extended Abstract). In *25th Annual Symposium on Foundations of Computer Science*, pages 464–479. IEEE, 1984.
6. David M. Goldschlag and Stuart G. Stubblebine. Publicly Verifiable Lotteries: Applications of Delaying Functions. In *Financial Cryptography 98*, LNCS 1465, pages 214–226. Springer, 1998.
7. C. Hall and B. Schneier. Remote Electronic Gambling. In *13th Annual Computer Security Applications Conference, ACM Press*, pages 227–230, December 1997.
8. Wooseok Ham and Kwangjo Kim. A Secure On-line Lottery Using Bank as a Notary. In *CISC 02*, pages 121–124.
9. K. Kobayashi, H. Morita, M. Hakuta, and T. Nakanowatari. An Electronic Soccer Lottery System that Uses Bit Commitment. *IEICE Transactions on Information and Systems*, E83-D(5):980–987, 2000.
10. Elisavet Konstantinou, Vasiliki Liagkou, Paul Spirakis, Yannis C. Stamatiou, and Moti Yung. Electronic National Lotteries. In *Financial Cryptography 04*, LNCS 3110. Springer, 2004.
11. Eyal Kushilevitz and Tal Rabin. Fair e-Lotteries and e-Casinos. In *CT-RSA 2001, The Cryptographer's Track at RSA Conference*, LNCS 2020, pages 100–109, 2001.
12. Anna Lysyanskaya. Unique Signatures and Verifiable Random Functions from the DH-DDH Separation. In *CRYPTO 02*, LNCS 2442, pages 597–612. Springer, 2002.
13. Silvio Micali, Michael O. Rabin, and Salil P. Vadhan. Verifiable Random Functions. In *IEEE Symposium on Foundations of Computer Science*, pages 120–130, 1999.
14. Kazue Sako. Implementation of a Digital Lottery Server on WWW. In *Secure Networking - CQRE (Secure) 99*, LNCS 1740, pages 101–108. Springer, 1999.
15. Jianying Zhou and Chunfu Tan. Playing Lottery on the Internet. In *Information and Communications Security - ICICS 01*, LNCS 2229, pages 189–201. Springer.