A Survey on e-voting protocols based on secret sharing techniques

Wafa Neji[*] — Kaouther Blibech[**] — Narjes Ben Rajeb [***]

[*] Higher Institute of Technological Studies of Beja, The General Directorate of Technological Studies, Tunisia
wafa.neji@gmail.com

[**] Higher Institute of the Environment, Town planning and Building Technologies, University of Carthage, Tunisia
kaouther.blibech@gmail.com

[***] National Institute of Applied Sciences and Technology, University of Carthage, Tunisia
narjes.benrajeb@gmail.com

ABSTRACT. Secret sharing techniques allow sharing a secret between a group of participants such that each of them holds one part of it. The secret can be reconstructed only when a subset of valid shares are combined together. These techniques are used in e-voting protocols since they allow to distribute the trust between several authorities and thus, achieve a greater degree of security. In this paper, we propose a classification of existing e-voting protocols based on secret sharing techniques and analyze their main advantages and drawbacks. We also identify security properties that could be ensured when using secret sharing techniques.

RÉSUMÉ. Le partage de secrets permet de partager un secret entre un ensemble de participants, chacun d'entre eux disposant d'une part du secret. Le secret ne peut être reconstruit que quand un sous-ensemble de parts valides sont réunies. Ces techniques sont utilisées dans les protocoles de vote électronique afin de distribuer la confiance entre plusieurs autorités électorales et atteindre ainsi un degré de sécurité plus important. Dans ce papier, nous proposons une classification des protocoles de vote électronique préexistants basés sur les techniques de partage de secret et nous analysons leurs avantages et inconvénients. Nous déterminons aussi les exigences de sécurité qui pourraient être satisfaites grâce à l'utilisation des différentes techniques de partage de secret.

KEYWORDS : Electronic voting, security, secret sharing, distributed key generation protocol

MOTS-CLÉS : Vote électronique, sécurité, partage de secret, protocole de génération de clé distribuée

1. Introduction

Electronic voting protocols are based on different approaches and cryptographic mechanisms that allow them to guarantee the validity of the voting process. Secret sharing techniques are one of the commonly used approaches because they avoid that a single electoral authority has the power to decrypt individual ballots, to access partial result, or to compute exclusively the final result. These behaviors compromise the security requirements of the voting process [1, 2]. Indeed, during an election, e-voting protocols must satisfy security properties of voting process. Foremost, all votes must be kept secret (privacy), and no traceability between the voter and his vote can be established (anonymity). Moreover, anyone can check the validity of the final voting result and voters must be able to ensure that their votes have been taken into account (verifiability) while preventing them from proving for any party how they voted (receipt-freeness). In addition, voters must also be able to vote correctly even if they are under a threat of an adversary (Incoercibility). Furthermore, the protocol must be robust against a coalition of a partial number of dishonest authorities (robustness).The complexity of the protocol is an important element which must be taken into consideration. Indeed, an efficient e-voting protocol has to be scalable according to time, communication and computation costs needed to include a larger number of voters (scalability).

Note that the use of cryptographic mechanisms in e-voting protocol could contribute to ensure these security requirements. Several comparative studies of these mechanisms have been proposed in the literature [1, 2, 3, 4]. Most of these works only provide an overview of these approaches and a basic understanding of e-voting protocols. Security requirements are not studied according to the used cryptographic techniques.

In this paper, we propose a particular analysis of e-voting protocols based on the used Secret Sharing Techniques (SST). In addition, we provide the security requirements that could be ensured thanks to the use of SST techniques. First, we introduce the notion of SST. Second, we present a classification of e-voting protocols according to the used SST. After that, based on this classification, we analyze the main advantages and drawbacks of e-voting protocols based on SST and we identify the most important security requirements that could be ensured through that.

2. Secret sharing techniques

The secret sharing is a cryptographic mechanism that allows to divide a secret data s, chosen initially by a trusted party named the dealer, in several parts $s_1, ..., s_n$. These shares are distributed among n participants such that only the coalition of a subset t of them allows the reconstruction of the original secret s. This mechanism is called the (t, n)

threshold secret sharing scheme. The first secret sharing schemes appeared in 1970 and were proposed simultaneously by Shamir [5] and Blakley [6]. The main problem of these schemes is that there is no guarantee of an adequate reconstruction of the secret if the dealer cheats by generating and distributing invalid shares or if one of the participants cheats by restoring invalid shares. Verifiable Secret Sharing (VSS) schemes [7] partially solve this problem because they allow participants to verify the validity of the shares received from the dealer. Unfortunately, dishonest participant can still restore invalid shares and skew the reconstruction of the secret. The solution of this problem is provided by Publicly Verifiable Secret Sharing (PVSS) schemes that allow not only the participants but also any external party to verify the validity of the distributed and/or the restituted shares. In the literature, several PVSS schemes has been proposed [8, 9, 10, 11, 12, 13]. These schemes can be used as a building block to design secure e-voting protocols.

3. Classification of e-voting protocols based on SST

In e-voting protocols, when a single authority supports the execution of the whole voting process, it is not possible to guarantee security requirements of the protocol. For example, if a private secret key used to decrypt ballots is owned by a single authority, this latter can decrypt voters' ballots and know the vote of each voter. To avoid this kind of situation, the secret key should not be held by a single authority and must be shared among several authorities. This process distributes the trust to achieve a greater degree of security, and reduces the risk of the presence of any dishonest authorities. During the voting process, secret sharing can be used in three different ways, as follows:

1) Class 1 : The secret is a private key shared between authorities. This private key is used to decrypt all ballots.

2) Class 2 :The secret is a ballot. Each voter uses a secret sharing scheme to share its ballot between authorities.

3) Class 3 :The secret is a decryption key of a single ballot. Each voter uses a secret sharing scheme to share the decryption key of its ballot between authorities.

Based on these three approaches, we propose in what follows a classification of e-voting protocols using SST.

3.1. Class 1: Authorities' shared key

For this first class, SST are used before the beginning of the voting process. During the initialization phase of the election, a trusted party generates and shares a private key between authorities. The public key associated with this private key is used by voters to encrypt their ballots. The secret key is used by authorities during the tallying phase to compute the final voting result. The use of this technique appeared in [14] and has been improved in [15, 16, 17, 18, 19, 20, 21, 22, 23, 24]. In general case, the generation of the secret key is carried out using secret sharing schemes [17, 18, 19, 20]. In the literature, a multitude of e-voting protocols of Class 1 use Shamir's secret sharing scheme as a building block. The major disadvantage of this is to involve a single trusted dealer who initially holds the secret key. In fact, this latter can decrypt individual ballots and compute partial result of final tallying. This compromises the security of the voting process. Another disadvantage is the lack of verification protocols for distributed and/or restituted shares. Thus, it is more appropriate to use VSS or PVSS schemes that include verification protocols. This ensures the validity of the shares distributed by the dealer, and restituted by the authorities. To avoid the need for a single trusted distributor, several e-voting protocols use distributed key generation protocols (DKG) which involve multiple parties to jointly generate and share the secret key. A multitude of protocols defined in the literature [14, 15, 16, 21] are based on the DKG protocol of Pedersen [25]. However, this DKG protocol has some drawbacks. It has been proven in [26] that this protocol cannot ensure a uniform distribution of the generated keys. Thus, the use of a secure DKG protocol to define e-voting protocols belonging to Class 1 is essential. Several authors proposes protocols of the Class 1 [14, 15, 18, 16, 19, 20, 21] which are based on secure DKG protocols. These protocols propose new versions of threshold cryptosystems used to encrypt ballots. The use of threshold cryptosystems is useful: the authorities cooperate to perform a multiple decryption of the final result without decrypting the ballots one by one. In addition, the secret key of the authorities is never reconstructed and is used implicitly in the tallying phase.

3.2. Class 2: Shared ballot

For this class, the SST are used during the voting process. Each voter acts as a dealer and shares its ballot between authorities using a secret sharing scheme. Each authority receives a different part of each ballot. To compute the final voting result, authorities use the homomorphic property [28] of the secret sharing scheme and multiply all the shares received from voters. In the literature, a multitude of e-voting protocols use this technique. E-voting protocols of Class 2 appeared first in [27] and were later improved in [28, 30, 31, 32, 33]. However, these protocols have some drawbacks. On the one hand, the majority of them, like those proposed in [30, 31, 33], struggle to prove the validity of the voting value contained in the shared ballot. On the other hand, some of these protocols [30, 31, 33] use non-verifiable secret sharing schemes as a building block. Thus, these schemes don't provide means to check the validity of ballots' shares distributed by voters and restituted by authorities. Voters can send invalid shares of their ballots and authorities can falsify the voting result by giving shares that do not actually come from the voters. For this purpose, protocols of Class 2 based on non-verifiable secret sharing schemes cannot ensure the verifiability property. The solution to this problem can be provided using VSS or PVSS schemes, which add verification protocols to check the validity of ballots' shares. This is the case, for example, of the voting protocol proposed by Cramer et al. in [28] which is based on the VSS scheme of Pedersen [29]. Unfortunately, the use of VSS or PVSS schemes is not possible in all cases. This is due to the use of verification protocols that compromise the confidentiality of the vote. This is the case, for example, of Feldman's VSS scheme, in which the content of the vote can be revealed from the public commitments of the Shamir polynomial coefficients using a simple exhaustive search (since each vote belongs to a predefined set of values). The use of Pedersen's VSS scheme [29] is more appropriate in this case since the public commitments used to verify the shares do not provide any information on the value of the vote. For this purpose, the most appropriate secret-sharing schemes for protocols of Class 2 are VSS or PVSS schemes which preserve the confidentiality of the vote and which have an homomorphic property facilitating the computing of the final voting result.

3.3. Class 3: Ballot's shared key

Electronic voting protocols in this class use a similar technique to the one used in the protocols of the second class. However, instead of sharing his/her ballot, each voter will share a secret key between authorities. This secret key is used by the voter to encrypt the ballot. Then, the coalition of authorities is needed to reconstruct the secret key of each voter and to decrypt ballots. The first e-voting protocol that uses this technique was proposed by Schoenmakers in 1999. It's also is the first e-voting protocol based on a PVSS scheme. In 2002, Kiayias and Yung [34] took inspiration from Schoenmakers' protocol to propose a protocol allowing voters to participate in the tallying phase to compute the final voting result. However, this protocol is based on (m, m) secret sharing scheme (where m is the number of voters) and requires the presence of all voters to compute the final result. In 2014, Zou et al. also propose in [35] an e-voting protocol based on (m, m) secret sharing scheme. The major disadvantage of this approach is that if only one of the shares is lost, the voting result will be permanently inaccessible. Moreover, the time and complexity of communication defined in [34, 35] depend on the number of voters. These protocols can be applied only to elections with a small number of voters.

4. Analysis of classes' security requirements

In this section, we focus our analysis on the security requirement that could be ensured by voting protocols thanks to the use of SST.

4.1. Privacy

The use of SST helps to satisfy this security requirement. For protocols belonging in each class, violating the privacy of a ballot implies that an adversary can compute the secret key shared between authorities (Class 1), can reconstruct the ballot from the shares sent to authorities (Class 2), or can compute the secret key related to the ballot from shares sent to authorities (Class 3). In all these cases, compromising the vote's privacy implies getting a secret data shared between authorities with a secret sharing scheme. Therefore, privacy of ballots depends on the security of the secret sharing scheme which must satisfy the secret property. Note that a secret sharing scheme satisfies the secret property for a secret data s if a dishonest party cannot get s, or any information related to s. Thus, if the secret property is satisfied, a dishonest party cannot decrypt ballots and cannot get any information related to the votes' values. This helps to ensure the privacy of the vote.

Remark 1. *The use of a secret sharing scheme that verifies the secret property contributes to ensure the privacy of the vote.*

For this purpose, e-voting protocols based on SST satisfy the privacy if the used secret sharing scheme verifies the secret property. Note that secret sharing schemes use several cryptographic primitives to ensure secrecy. Most of these primitives are based on NP-hard problems.

4.2. Anonymity

E-voting protocols belonging to Classes 1, 2 and 3 satisfy voter's anonymity by assuming the honesty of a subset of authorities who will not cooperate to decrypt individual ballots. For this purpose, this assumes that authorities will not cooperate to explicitly reconstruct the authorities'secret key (Class 1), or will not cooperate to reconstitute individual ballots from received shares (Class 2), or will not explicitly reconstitute the secret key related to each ballot (Class 3).

Remark 2. *Voter's anonymity could be satisfied only assuming the honesty of at least t of the authorities who will not cooperate to decrypt ballots one by one.*

4.3. Receipt-Freeness

E-voting protocols based on STT in Class 2 and Class 3 fail to satisfy receipt-Freeness. This is due to the random values chosen by the voter to share his ballot (Class 2), or to share the secret key related to his ballot (C lass3). Thus, a voter can construct a receipt which can prove the content of his vote by revealing the random value that he used during the dealing phase. In general, in a voting process, when the voter chooses a random value to encrypt his vote, the voter can easily use it to construct a receipt of his vote [15, 16].

Remark 3. *Receipt-Freeness is not satisfied by e-voting protocols based on SST used in Class 2 and Class 3.*

In e-voting protocol of Class 1, the voter does not execute the secret sharing process to encrypt his ballot. He only uses the authorities'secret key. If he has to choose in addition a random value it is possible to re-encrypt the value of encrypted ballot. The re-encryption can be performed using re-encryption mix-net [36], permutations carried out by authorities or by using a secure hardware device [16]. This process prevents the voter from keeping chosen random values. The voter is then unable to prove to an adversary the content of his vote

Remark 4. *Receipt-Freeness can be satisfied by e-voting protocols based on SST used in Class1 and combined with mix-net or re-encryption technique.*

Note that the combination of these techniques has some disadvantages. The use of re-encryption technique implies adding new proofs of verification to prove the validity of the re-encryption. In the case of using of re-encryption mix-nets, this leads to an important communication and computational complexity. New validity proofs must be added in each stage.

4.4. Incoercibility

In e-voting protocols based only on SST, it is not possible to satisfy the incoercibility. To avoid this, it is possible to resort to the use of anonymous credentials in combination with SST. Indeed, for the protocols of Class 1 and Class 3, to vote, each voter must submit a credential with his encrypted ballot to validate it. For protocols of Class 2, during the dealing phase, each voter must submit several credentials with the ballot shares to validate them. In any case, if an adversary forces the voter to vote in a certain way, the voter may submit an invalid credential. Recall that neither the voter nor the attacker can prove or verify the validity nor the invalidity of the submitted credential.

Remark 5. *The combination of SST with anonymous credentials contribute to ensure incoercibility.*

4.5. Robustness

The robustness implies that the voting protocol can tolerate the presence of a number of dishonest authorities. E-voting protocols based on SST can ensure this requirement. These protocols assume the presence of a minimal number t of honest authorities to share the authorities' secret key (Class 1), the ballots (Class 2) or the secret keys related to the ballots (Class 3). Inadequate behavior of $t - 1$ coalition of authorities can be tolerated. No coalition of dishonest voters can disrupt the election.

Remark 6. *The use of (t, n) threshold secret sharing schemes in e-voting protocol contributes to satisfy the robustness property.*

4.6. Verifiability

In PVSS schemes, a public validity proof is added to allow any party to verify the validity of the distributed and restored shares. The application of a PVSS scheme to an e-voting protocol ensures verifiability. In fact, any participant can ensure that the ballots have been counted correctly by verifying the validity of ballots' shares (Class 2) or decryption keys' shares (Class 3) distributed by voters and given back by authorities. The decryption that leads to the final result from valid ballots is also verifiable. Note also that the use of a PVSS scheme or a DKG protocol based on PVSS scheme for protocols of Class 1 also allows any participant to verify the validity of the distributed decryption performed by authorities.

Remark 7. *The use of SST based on PVSS schema could contribute to ensure verifiability.*

4.7. Scalability

When examining the performance of the voting process, we notice that the work done by the voter in protocols of Class 1 seems requiring less computation operations than Class 2 and Class 3 [14]. An interesting property of protocols of Class 1 is to see whether that complexity and communication time are independent of the number of voters and authorities. Indeed, in protocols of Class 1, a voter will simply send a single encrypted ballot accompanied by a single proof that proves the validity of his vote. Nevertheless, in protocols of Classes 2 and 3, the voter must send several encrypted shares according to the number of authorities and must prove the validity of each share.

Remark 8. *The property of scalability could be provided by e-voting protocols of Class 1.*

For this purpose, protocols belonging to Classes 2 and 3 seem to be more appropriate for small elections, on account of the complexity of computational operations made during the voting and tallying phases and the manifold proofs generated by voters. The protocols belonging to Class 1 can be applied for large-scale elections.

4.8. Summary of the analysis

Table 1 provides a summary of the security requirements that could be satisfied by each class. From the Table 1, we deduce that the use of the secret sharing technique related to Class 1 is the most appropriate for the design and the definition of e-voting protocols. Combined with other cryptographic approaches (anonymous credentials, re-encryption, mix-nets, etc.), this technique helps to satisfy the security requirements of the voting process. Note that the most appropriate way to exploit this technique is using DKG protocols based on PVSS schemes. Indeed, on the one hand, this avoid having recourse to a trusted party who initially holds the secret key, and on the other hand contributes to ensure the property of verifiability.

Table 1 – Classification: Analysis of security requirements

	Privacy		Robust-	Receipt-	Incoercibil-	Verifiability	Scalability
		Anonymity	ness	freeness	ity		
Class 1	Com	C	✓	CCA	CCA	PVSS / DKG based on PVSS	✓
Class 2	Com	C	✓	X	CCA	PVSS	X
Class 3	Com	C	✓	X	CCA	PVSS	X

Com : computational privacy, **C** : conditionally satisfied, **CCA** combination with other cryptographic approaches, ✓ : satisfied, **X** : not satisfied

5. Conclusion

In this paper, we have studied the use of SST in e-voting protocols and analyze their main advantages and drawbacks. We have also proposed a classification of e-voting protocols based on the used SST. This classification led us to identify security properties that could be satisfied for each class. Depending on the targeted security requirements, our analysis may help in the selection of building blocks and cryptographic mechanisms that could be used in order to define secure electronic voting protocols.

It should be noted that the use of specific cryptographic approaches does not necessarily imply that e-voting protocols satisfy the required properties of e-voting process. For each protocol, it should be necessary to verify that the combination of all cryptographic building blocks contributes to ensure security requirements. Indeed, it would be interesting to formally prove the security of e-voting protocols. Thus, as future research, we intend to construct formal proofs in order to prove the satisfaction of security requirements related to e-voting protocols.

6. References

[1] Mursi, M. F. , Assassa, G. M. , Abdelhafez, A. , Samra, K. M. A., “ On the development of electronic voting: a survey ”, *International Journal of Computer Applications*, Vol. 61, no 16, 2013.

[2] Qadah, G. Z. , Taha, R., “ Electronic voting systems: Requirements, design, and implementation ”, *Computer Standards and Interfaces*, Vol. 29, no 3, 376-386, 2007.

[3] Lambrinoudakis, C. , Gritzalis, D. , Tsoumas, V. , Karyda, M. , Ikonomopoulos, S., “ Secure electronic voting: The current landscape”, *In Secure electronic voting, Springer, Boston*, 101-122, 2003.

[4] SMITH, W. D., “Cryptography meets voting”, *Technical report*, Vol. 10, 80, 2005.

[5] SHAMIR, A, “How to share a secret”, *Communications of the ACM*, Vol. 22, no 11, 612-613, 1979.

[6] BLAKLEY, G. R., “Safeguarding cryptographic keys”, *In Proceedings of the national computer conference*, Vol. 48, 313-317, 1979.

[7] FELDMAN, P., “ A practical scheme for non-interactive verifiable secret sharing”, *28th Annual Symposium on. IEEE*, 427-438, 1987.

[8] BEHNAD, A. , EGHLIDOS, T., “A new, publicly verifiable, secret sharing scheme”, *Sci. Iran.* , 246-251, 2008.

[9] SCHOENMAKERS, B., “A simple publicly verifiable secret sharing scheme and its application to electronic voting”, *In Annual International Cryptology Conference, Springer, Berlin, Heidelberg*, 148-164, 1999.

[10] FUJISAKI, F. , OKAMOTO, T., “A practical and provably secure scheme for publicly verifiable secret sharing and its applications”, *In: Proceedings of the annual international conference on Theory and application of cryptographic techniques, EUROCRYPT'98, Springer-Verlag, Berlin, Heidelberg*, 32-46, 1998.

[11] Behnad08,Schoenmakers99,Fujisaki,Heidarvand09 HEIDARVAND, S. , VILLAR, J. L., “Selected Areas in Cryptography”, *chapter Public Verifiability from Pairings in Secret Sharing Schemes, Springer-Verlag, Berlin, Heidelberg*, 294-308, 2009.

[12] JHANWAR, M. P., “A Practical (Non-interactive) Publicly Verifiable Secret Sharing Scheme”, *In: ISPEC'11*, 273-287, 2011.

[13] SHIL, A. , BLIBECH, K. , ROBBANA, R. , NEJI, W., “ A New PVSS Scheme with a Simple Encryption Function”, *arXiv preprint arXiv:1307.8209*, 2013.

[14] CRAMER, R. , GENNARO, R. , SCHOENMAKERS, B. , “A secure and optimally efficient multi-authority election scheme”, *Transactions on Emerging Telecommunications Technologies*, 8(5), 481-490, 1997.

[15] HIRT, M. , SAKO, K. , “Efficient receipt-free voting based on homomorphic encryption”, *In International Conference on the Theory and Applications of Cryptographic Techniques* , 539-556, 2000.

[16] LEE, B. , KIM, K., “Receipt-free electronic voting scheme with a tamper-resistant randomizer”, *In International Conference on Information Security and Cryptology*, 389-406, 2002.

[17] DAMGARD, I., “A generalisation, a simplification and some applications of Paillier's probabilistic public-key system”, *Proceedings of the 4th International Workshop on Practice and Theory in Public Key Cryptosystems*, 119-136, 2001.

[18] FOUQUE, P. A. , STERN, J., “One round threshold discrete-log key generation without private channels”, *Public Key Cryptography*, 300-316, 2001.

[19] ACQUISTI, A. , “Receipt-Free Homomorphic Elections and Write-in Ballots”, *IACR Cryptology ePrint Archive*, p 105, 2004.

[20] PORKODI, C. , ARUMUGANATHAN, R. , VIDYA, K., “Multi-authority Electronic Voting Scheme Based on Elliptic Curves”, *IJ Network Security*, Vol 12(2), 2011.

[21] PHILIP, A. A. , SIMON, S. A. , OLUREMI, A., " A receipt-free multi-authority e-voting system", *International Journal of Computer Applications*, Vol 30, no 6, 15-23, 2011.

[22] CHONDROS, N. , ZHANG, B. , ZACHARIAS, T. , DIAMANTOPOULOS, P. , MANEAS, S. , PATSONAKIS, C. , ROUSSOPOULOS, M., "D-DEMOS: A distributed, end-to-end verifiable, internet voting system", *In Distributed Computing Systems (ICDCS), 2016 IEEE 36th International Conference*, 711-720, 2016

[23] CULNANE, C. , RYAN, P. Y. , SCHNEIDER, S. , TEAGUE, V. , "vVote: a verifiable voting system", *ACM Transactions on Information and System Security (TISSEC)*, Vol 18, no 1, 2015.

[24] CHAIDOS, P. , CORTIER, V. , FUCHSBAUER, G. , GALINDO, D. , "Beleniosrf: A non-interactive receipt-free electronic voting scheme", *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 1614–1625, 2016.

[25] PEDERSEN, T. P., "A threshold cryptosystem without a trusted party", *In Workshop on the Theory and Application of Cryptographic Techniques*, 522-526, 1991.

[26] GENNARO, R. , JARECKI, S. , KRAWCZYK, H. , "Secure distributed key generation for discrete-log based cryptosystems", *Journal of Cryptology*, 20(1), p. 51-83, 2007.

[27] BENALOH, J. D. C. , " Verifiable secret-ballot elections", 1987.

[28] CRAMER, R. , FRANKLIN, M. , SCHOENMAKERS, B. , YUNG, M. , "Multi-authority secret-ballot elections with linear work", *In International Conference on the Theory and Applications of Cryptographic Techniques*, 72-83, 1996.

[29] PEDERSEN, T. P., "Non-interactive and information-theoretic secure verifiable secret sharing", *In Annual International Cryptology Conference*, 129-140, 1991.

[30] IFTENE, S., "General secret sharing based on the chinese remainder theorem with applications in e-voting", *Electronic Notes in Theoretical Computer Science*, 186, 67-84, 2007.

[31] SPIRIDONICĂ, A. M. , PISLARU, M., "THE ASSURANCE OF SECURITY OF ELECTRONIC VOTING THROUGH THE USE OF SECRETS SHARING SCHEMES AND BENALOH ELECTRONIC VOTING SCHEME", 2010.

[32] OTSUKA, A. , IMAI, H. , "Unconditionally secure electronic voting", *In Towards Trustworthy Elections*, 107-123, 2010.

[33] NAIR, D. G. , BINU, V. P. , KUMAR, G. S. , "An improved e-voting scheme using secret sharing based secure multi-party computation", *arXiv preprint arXiv:1502.07469*, 2015.

[34] KIAYIAS, A. , YUNG, M., "Self-tallying elections and perfect ballot secrecy", *International Workshop on Public Key Cryptography*, 141-18, 2002.

[35] ZOU, X. , LI, H. , SUI, Y. , PENG, W. , LI, F., "Assurable, transparent, and mutual restraining e-voting involving multiple conflicting parties", *INFOCOM, 2014 Proceedings IEEE*, 136-144, 2014.

[36] NEFF, C. A., "A verifiable secret shuffle and its application to e-voting", *Proceedings of the 8th ACM conference on Computer and Communications Security*, 116-125, 2001.

Appendix - Some existing e-voting proyocols : Analysis of security requirements

The Table 2, Table 3 and Table 4, give respectively a summary of the security requirements satisfied by some e-voting protocols belonging to Class 1, Class2 and Class3.

Table 2 – Class 1: Analysis of security requirements

Protocol	Privacy	Anonymity	Robustness	Receipt-freeness	Incoercibility	Verifiability	Scalability
Cramer et al. (1997)	Com	C	✓	X	X	✓	✓
Damgard et Jurik (2000)	Com	C	✓	X	X	✓	✓
Hirt et Sako (2000)	Com	C	✓	✓	X	✓	X
Fouque et al. (2001)	Com	C	✓	✓	X	✓	✓
Lee et Kim (2002)	Com	C	✓	C	X	✓	✓
Acquisti (2004)	Com	C	✓	AP	AP	X	X
Civitas/JCJ (2008)	Com	C	✓	✓	✓	✓	X
Porkodi et al. (2011)	Com	C	✓	X	X	✓	✓
Philip et al. (2011)	Com	C	✓	✓	X	✓	X
Chondros et al. (2015)	Com	C	✓	C	X	✓	✓
BeleniosRF (2016)	Com	C	✓	✓	✓	✓	✓

Com : computational privacy, **C** : conditionally satisfied, **AP** : attack proved, ✓ : satisfied, **X** : not satisfied

Table 3 – Class 2: Analysis of security requirements

Protocol	Privacy	Anonymity	Robustness	Receipt-freeness	Incoercibility	Verifiability	Scalability
Cramer et al. (1996)	Com	C	✓	X	X	✓	X
Iftene (2007)	Com	C	✓	X	X	X	X
Spiridoncia et al. (2010)	Com	C	✓	X	X	X	X
Otsku et Imai (2010)	Com	C	✓	X	X	✓	X
Mukhopadhyay (2014)	Com	C	X	X	X	X	X
Nair et al. (2015)	Com	C	✓	X	X	X	X

Com : computational privacy, **C** : conditionally satisfied ,✓ : satisfied, **X** : not satisfied

Table 4 – Class 3: Analysis of security requirements

Protocol	Privacy	Anonymity	Robustness	Receipt-freeness	Incoercibility	Verifiability	Scalability
Schoenmakers (1999)	Com	C	✓	X	X	✓	X
Kiayias et Yung (2002)	Com	C	X	X	X	✓	X
Zou et al. (2014)	Com	C	X	X	X	✓	X

Com : computational privacy, **C** : conditionally satisfied, ✓ : satisfied, **X** : not satisfied