INTERNATIONAL JOURNAL OF SOCIAL Sciences and Information Technology
ISSN 2412-0294 www.ijssit.com

http://www.ijssit.com

A REVIEW OF CRYPTOGRAPHIC PRIMITIVES FOR SECURITY OF ELECTRONIC VOTING SYSTEMS

[1*] **Jane Juma**
jjumacloy@gmail.com

[2**] **Charles Ochieng Oguk**
ogukcharles@gmail.com

[1,2] *Department of Mathematics, Statistics and Computer Science, School of Science Technology and Engineering, Rongo University*

Abstract*: Security around electronic systems is not only sensitive, but also remains contentious among key election stake-holders. Various cryptographic primitives have been adopted in e-voting systems to infuse confidentiality, integrity and availability, in order to entrench trust among to all stakeholders. However, the cryptographic techniques have not been well understood by the voters, candidates as well as other key election stake-holders. Presenting the mode of operation, the extent of providing security as well as the limitations of individual cryptographic primitive goes a long way to ensure that the security techniques are well understood by all stakeholders. This review looked into various cryptographic primitives applied in e-voting systems, how they achieve security, their strengths and limitations.*

Keywords*: cryptographic primitives, Encryption, Electronic voting systems, encryption, cryptography, voter privacy*

Introduction

Many security approaches have been applied at various electoral levels in different areas of e-voting systems. The security approaches have been results of application of various cryptographic approaches, which mainly convert the vote information as plain text into intelligible cipher texts, (Quaglia & Smyth, 2018). The cryptographic primitives applied in the most recent electronic voting systems have been: digital signatures, optical scan technology, Blind signatures, homomorphic cryptography, block-chain, El-Gamal cryptographic system, Elliptic Curve cryptography (ECC), public key cryptography and El-Gamal cryptographic system. These techniques have majorly been used to handle voter verification, vote count, and tallying of results. However, there is insufficient understanding of how the cryptographic primitives affect security in e-voting systems.

The researchers explain basic terms in cryptographic primitives as follows;

Plain Text: The original message or information about the choices that the voter wishes to communicate in the electronic voting system. In cryptography the actual voters' choice in the form of information at the polling is given a special name as Plain Text. For example, a student voter wishes to send “position n=candidate m” message to the system. Here “position n=candidate m” is a plain text message.

Cipher Text: The message that remains meaningless to the unintended party or system is what we call as Cipher Text. Through cryptography, the plain text message is transformed into cipher text before the transmission.

For example, "Akdhsdssdujk526#@91ukl8*^&%$#@%" could be a cipher text produced for "position n=candidate m".

Encryption : According to Kiayias, Zacharias, and Zhang, (2017), encryption is a process of achieving cipher text from plain text is called encryption technique. Various encryption techniques are used to send confidential messages through an insecure channel. Encryption process requires two things, which are an encryption algorithm and a key. An encryption algorithm is the program that converts plain text into cipher text. Encryption happens at the sender side. The general purpose of encryption is to provide confidentiality, authentication, integrity, non repudiation and access control for the critical data.

Decryption: A reverse process of encryption that involves reverting the cipher text into the plaintext at the receiver end is called as decryption. This process requires a decryption algorithm and a key. Generally, the encryption algorithm and decryption algorithm are same.

Key: a key is a numeric, alphabetical, alpha - numeric text or a combination of all the three and a special symbol; that is used at the time of encryption and also at the time of decryption, (Abandah, Darabkh, Ammari, & Qunsul, 2014).

Broadly, encryption algorithms are classified into two categories: Symmetric and Asymmetric key encryption. In symmetric key cryptography, a similar key used for encryption is as well used in decryption. In asymmetric key cryptography, different key are used for both encryption and decryption, (Kiayias, Zacharias, & Zhang, 2017a).

Statement of the Problem

While most of the modern e-voting systems have applied the principle of Digital Signatures, Optical Scan Technology, Blind signatures and homomorphic cryptography, block-chain, El-Gamal cryptographic system, Elliptic Curves, public key cryptography that uses ECC and El-Gamal cryptographic system, these approaches have individual limitations that has made the resultant systems lacking the necessary electoral processes security. It is necessary to explore the strengths as well as the weaknesses of the applicable cryptographic primitives.

Objective

To explore the features of Cryptographic Primitives for Security of Electronic Voting Systems

Literature Review

The study reviewed Digital Signatures, Optical Scan Technology, Blind signatures, homomorphic cryptography, block-chain, and El-Gamal cryptographic system, Elliptic Curves, public key cryptography and El-Gamal cryptographic system. This included the individual security success areas and the limitations of each technique.

Digital Signatures.

Digital signature is a digital code which is generated and authenticated by public key cryptography, and is attached to an electronically transmitted document so that it helps to verify the documents' contents and identity of the sender's as well as providing proof of original and unmodified documentation, (Cruz & Kaji, 2017). In electronic voting, the message of the voters' choice is secured using digital signature, where digital signature uses the principle of public key cryptography. This means that the user has to acquire private and public key, while the receiver has to obtain the digital signature certificate as well.

Limitations: Even though this approach has proved secure for e-voting, it has various drawbacks as that limits its wide application. Digital signature approach requires parties to pay additional amount of money for the digital certificates at both the sender and the receiver ends. Further, the process of generation and verification of digital signature needs considerable amount of time. In a typical election scenario, the process of voting compels a frequent exchange of messages which practically compels the speed of communication to greatly reduce, (Dworkin, 2015) and this beats the essence of electronic voting.

In addition, If a user for example a voter changes his private key after every fixed interval of time, it implies then the records of all these changes made must be kept so that if a dispute arises over a previously sent message, then the old key pair needs to be referred, (Cruz & Kaji, 2017). Practically, the computing storage of all the previous keys in a practical election and voting processes is another overhead. In elections, secrecy is of paramount interest. However, while digital signature provides authenticity, it hardly ensures secrecy of the data,(Yin, Fu, & Chen, 2016). Therefore, to provide the secrecy, additional cryptographic technique needs to be considered for effective verifiable voting system. In order to smoothly use digital signatures, both voters and receiving systems have to buy digital certificates, which are expensively acquired from trusted certification authorities. Additional cost comes in the price of a special software that works with digital certificates, (Yin et al., 2016), which both senders and recipients of electoral messages have to buy at a cost.

Optical Scan Technology

An optical scan voting technology is hybrid voting system which uses optical scanner to read the already marked paper ballots and tally the results.

Limitations: This however, does not provide electronic voting mechanisms but is only an electronic relic of the problematic manual voting system.

Blind Signature

Blind signature is a type of digital signature, but in which the message is blinded just before it is signed, making it very difficult for the signer to access the message content, (López-García, Dominguez Perez, & Rodríguez-Henríquez, 2014). After signing, the signed message is then un-blinded. This makes it similar to a normal digital signature, and therefore, it can be publicly checked against the original message. Blind signature can as well be implemented using the several public-key cryptographic primitives. However, in practical voting where both security and verifiability is required, blind signatures are limited in a number of ways.

Limitations: Its simple blinded signature scheme depends on the operational assumptions that the server user gets no information on the message being signed once the message is received. However, as the server keeps receiving more information, things start being linkable and this jeopardizes secrecy, (Darwish & M El Gendy, 2017). Consequently, by keeping a log of blinded messages as the signing process goes on, it becomes easy for a signing party to link blind messages to their revealed un-blinded versions, and in this way, privacy is greatly compromised.

Further**,** most voting systems should allow duplicate voting to prevent coercion. However, blindly signed ballots do not present any connection to the original voter. Therefore, it is not possible to find other votes cast by the same voter in order to drop all except the last of the ballots, (Mateu, Miret, & Sebé, 2016). Consequently, this cryptographic primitive is only ideal in some theoretical schemes, but hardly finds applicability in real-world voting systems. As such, no known voting systems practically use blind signatures, since they all allow re-voting to override old ballots.

Homomorphic encryption

Homomorphism is a method of encryption that allows data to remain encrypted while it is being operated on, processed and manipulated, (Cortier, Eigner, Kremer, Maffei, & Wiedling, 2015) . Homomorphic encryption therefore allows computation on cipher texts, and thereby generating an encrypted result. When decrypted, such encrypted results match the result of the operations as if they had been performed on the plaintext.

Limitation: In practical perspective, this cryptographic primitive ranks among the worst cryptographic technique in terms of speed, (Xiang, Yu, & Zhu, 2012). According to Atzei, Bartoletti, and Cimoli, (2017), homomorphic encryption remains impractically slow, and with very high computational overheads. In elections with many voters, the systems may become so slow that the stake-holders may not appreciate the essence of electronic voting as compared to the already known slow manual voting approach. For example, speed became a great issue of concern during the project of making homomorphic encryption widespread by IBM, when it released its first version of HElib C++ library in the year 2016, (Helsloot, Tillem, & Erkin, 2017).

Furthermore, there are restrictions and usage in electronic voting systems. According to (Yu et al., 2018), using an encryption scheme dependent on homomorphic properties of allowing the manipulation of the cipher texts' variables, there is a restriction in the structure of the ballot. Before the ballots are encrypted, they must be encoded with bits. This means that the candidate whom the voter intends to vote for gets a one (1) whilst all other candidates have a zero (0) for every vote cast and this information is stored in the corresponding position, which is then encrypted. As a result of this structure, it can only accommodate elections where *yes* or *no* are possible choices for the candidates. Ordinary voting in elections is therefore not supported because we cannot encode a string involving the name of a candidate into one bit. Also given that the homomorphic addition does not support addition of strings, it is therefore not feasible. Another big limitation of this cryptographic primitive towards secure voting is the computing time needed to aggregate homomorphic encrypted ballots. This is really complex and as such, might not be applicable on large amounts of encrypted ballots.

Block chain technology

Block chain is a distributed database for transactions which exist on multiple computers at the same time, and is constantly growing as new sets of information pieces are getting updated, (Tarasov & Tewari, 2017). Each block of information contains a timestamp as well as a link to the previous block of information, thereby forming a chain of information on transactions. The original block chain technology offered an alternative to the traditional intermediary for transactions of the crypto currency Bitcoin. The collective verification process when applied in e-voting offers a huge degree of traceability and security in electronic elections, a principle that is applied in secure electronic voting, (Yu et al., 2018). This is because there are multiple versions of nodes on a network acting both as executors of transactions and miners simultaneously, whereas the voting processes (transactions) are collected into blocks before being added to the wider election systems' block chain.

Limitations: This technology comes with much complexity and involves an entirely new vocabulary beyond the comprehension of most election stake-holders, especially the voters, candidates and election managers, (Panja & Roy 2018). Due to this complexity and computing intensity, block chain technology requires supercomputers and similarly powerful hardware resources for each transaction, in order to cope with the heavy energy consumption and computing intensity associated with high data traffic in large elections. Also, Fusco, Lunesu, Pani and Pinna, (2018) showed that privacy remains an issue of concern, since despite the fact that the identities of those involved in I-voting process are anonymous, the technology presents patterns to the

transactions, and these patterns that can easily portray the identity of every user with a particular addresses, thus providing an easy way to extract information about the vote and the voter. In the log chain of information blocks, there is the risk of error due to human involvement in the generation of blocks of election information.

Further, voting systems developed from block chain technologies may suffer the lack of certified professionals for implementation and maintenance, (Chen et al., 2017). This is coupled with the fact that industry experts have revealed that this technology is not easy to learn, due to its sophisticated mathematical functions and complex algorithms. Scalability is a further limitation, since all the transactions have to be verified by the entire entire node in the network, thus limiting the speed of the transaction process and need for very high computing power especially in an election involving multiple voters from large geographical areas. Apart from the realization that block chain transactions are not immutable and not indestructible, there is a major security flaw in the technology due to a '51% attack'. This is means if more than half of the computers working as nodes in the network tell a lie, the lie will be promoted and hence become the truth, what is known as '51% attack', (Nakamoto, 2008).

El-Gamal cryptographic system

Vijayalakshmi and Karpagam (2018) explained this type of encryption as a public-key cryptography which uses asymmetric key encryption for communication between two parties and encrypting the message based on the Diffie–Hellman key exchange. It consists of three components: the *key generator*, the *encryption algorithm*, and finally the *decryption algorithm*. In cryptography, ElGamal encryption can be defined over any given cyclic group G, for example, a multiplicative group of integers modulo *n*. Its security therefore depends on the difficulty of a certain problem in cyclic group related to computing discrete logarithms. The three components of ElGamal encryption therefore ensure difficulty of finding discrete logarithm in the given cyclic group. In simple terms, given the variables g, a and k, then in a function where one knows g^a and g^k, the cryptographic primitive makes it extremely difficult to compute g^{ak}., and this provides the needed privacy, (Vijayalakshmi & Karpagam 2018)

Limitations: On the election security perspective, semantic security is not implied in this technique by the computational Diffie–Hellman assumption alone. This implies that the cryptosystem is unconditionally malleable, and this consequently makes it not secure for electronic voting under chosen cipher text attacks. According to Faust, Mukherjee, Venturi and Wichs, (2014) an encryption algorithm is "malleable" if it has a possibility to transform a cipher text into another cipher text which decrypts to the related plaintext. Malleability is undesirable property in a cryptosystem, since it allows an attacker to access the plain text and modify the contents of a message.

Distributed Systems and Cryptography

This involves distributed systems voting / election algorithm and distributed processing. Distributed Algorithm is a algorithm that runs on a distributed system, while distributed system is a collection of independent computing systems that do not share their memory, wherein each processor has its own memory and they only communicate via communication networks, (Sandler, Derr, & Wallach, 2008). Communication in this case is implemented in a processor where one machine communicates with a processor on other machine. Many algorithms applied in distributed systems require a coordinator that performs vital functions needed by other processes in the system. Similarly, election algorithms are designed to choose a coordinator.

Limitations of Election Algorithms: Since election algorithms choose a process from group of processors that act as a coordinator, whenever the coordinator process crashes, then a new coordinator is elected on other processor basically determines where a new copy of coordinator should be restarted. It operates on the assumption that every active process in the system has a unique priority number, wherein the process with highest priority will be chosen as a new coordinator, (Kho et al., 2015). Hence, in case of coordinator failure, the algorithm elects the active process with the highest priority number, which is then sent to every active process in the distributed system. The resultant major limitations with this technology are the inherent lack of global clock as well as lack of shared memory. These are counter to secure elections systems, (Chondros et al., 2019).

Elliptic-Curve Cryptography

Elliptic-curve cryptography is an approach to public-key cryptography defined over finite fields and is based on the algebraic structure of elliptic curves to provide data security equivalent to classical systems (like RSA), but uses fewer bits, smaller chip size, less power consumption but increase in speed (Jaiswal & Tripathi, 2017). In operation, Elliptic curve cryptography (ECC) applies the mathematical properties of elliptic curves to generate public key cryptographic systems. Similar to all public-key cryptography, the technique is based on mathematical functions which are simple to compute in one direction, but extremely difficult to reverse. The difficulty resides in the infeasibility to compute the discrete logarithm associated with any random elliptic curve element with regards to a publicly known base point, in the elliptic curve discrete logarithm problem.

Limitations: Despite the security offered by the cryptographic primitive, there are a significant number of potential vulnerabilities to elliptic curves which make them unsuitable for secure e-voting systems. Side-channel attacks and twist-security attacks threaten to invalidate the security that elliptic curve cryptography aims to provide, especially to private keys, (Chaieb, Yousfi, Lafourcade, & Robbana, 2019). Side-channel attacks are widely experienced when measurements are made on the practical implementation of a cryptosystem, but which often results into leaks of information, thus compromising secrecy and anonymity.

Further, incorrect implementation often leads to ECC private key leaks in a number of scenarios. For example, the Sony ECDSA security disaster, wherein, while Sony used ECDSA to sign software for their Play-Station game console, they however, did not implement the algorithm properly, and used static parameters instead of random ones thus making Sony's implementation of the algorithm solvable and subsequently useless, (Knezevic, Nikov, & Rombouts, 2016). Solvable encryption algorithm is unsuitable for secure e-voting systems. Further, key exchange message to malformed signatures have been experienced with ECC, and these become worse, as such issues often lead to an unauthenticated, internet based remote attacker gaining access to Secure Sockets Layer (SSL) private keys.

Recommendation and Conclusion

The cryptographic techniques above have strengths as well as weaknesses in providing confidentiality, integrity, availability and general security necessary in e-voting. It is recommended that for secure e-voting system to be achieved, positive aspects of all the cryptographic primitives need to be harnessed to develop a hybrid and really secure system.

References

Abandah, G. A., Darabkh, K. A., Ammari, T., & Qunsul, O. (2014). Secure national electronic voting system. Journal of Information Science and Engineering.

Atzei, N., Bartoletti, M., & Cimoli, T. (2017). A survey of attacks on Ethereum smart contracts (SoK). Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/978-3-662-54455-6_8

Chaieb, M., Yousfi, S., Lafourcade, P., & Robbana, R. (2019). Verify-your-vote: A verifiable blockchain-based online voting protocol. Lecture Notes in Business Information Processing. https://doi.org/10.1007/978-3-030-11395-7_2

Chen, S., Shi, R., Ren, Z., Yan, J., Shi, Y., & Zhang, J. (2017). A Blockchain-Based Supply Chain Quality Management Framework. Proceedings - 14th IEEE International Conference on E-Business Engineering, ICEBE 2017 - Including 13th Workshop on Service-Oriented Applications, Integration and Collaboration, SOAIC 207. https://doi.org/10.1109/ICEBE.2017.34

Chondros, N., Zhang, B., Zacharias, T., Diamantopoulos, P., Maneas, S., Patsonakis, C., Roussopoulos, M. (2019). Distributed, end-to-end verifiable, and privacy-preserving internet voting systems. Computers and Security, 83(August), 268–299. https://doi.org/10.1016/j.cose.2019.03.001

Cortier, V., Eigner, F., Kremer, S., Maffei, M., & Wiedling, C. (2015). Principles of Security and Trust. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/978-3-662-46666-7

Cruz, J. P., & Kaji, Y. (2017). E - voting System Based on the Bitcoin Protocol and Blind Signatures. IPSJ Transactions on Mathematical Modeling and Its Applications.

Darwish, A., & M El Gendy, M. (2017). A New Cryptographic Voting Verifiable Scheme for E-Voting System Based on Bit Commitment and Blind Signature. International Journal of Swarm Intelligence and Evolutionary Computation. https://doi.org/10.4172/2090-4908.1000158

Dworkin, M. J. (2015). SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions. In Draft FIPS PUB 202. https://doi.org/10.6028/NIST.FIPS.202

Faust, S., Mukherjee, P., Venturi, D., & Wichs, D. (2014). Efficient non-malleable codes and key-derivation for poly-size tampering circuits. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/978-3-642-55220-5_7

Fusco, F., Lunesu, M. I., Pani, F. E., & Pinna, A. (2018). Crypto-voting, a Blockchain based e-Voting System. https://doi.org/10.5220/0006962102230227

Helsloot, L. J., Tillem, G., & Erkin, Z. (2017). AHEad: Privacy-preserving online behavioural advertising using homomorphic encryption. 2017 IEEE Workshop on Information Forensics and Security, WIFS 2017. https://doi.org/10.1109/WIFS.2017.8267662

Jaiswal, P., & Tripathi, S. (2017). An authenticated group key transfer protocol using elliptic curve cryptography. Peer-to-Peer Networking and Applications. https://doi.org/10.1007/s12083-016-0434-7

Kho, A. N., Cashy, J. P., Jackson, K. L., Pah, A. R., Goel, S., Boehnke, J., Galanter, W. L. (2015). Design and implementation of a privacy preserving electronic health record linkage tool in Chicago. Journal of the American Medical Informatics Association. https://doi.org/10.1093/jamia/ocv038

Kiayias, A., Zacharias, T., & Zhang, B. (2017a). An Efficient E2E Verifiable E-voting System without Setup Assumptions. IEEE Security and Privacy, 15(3), 14–23. https://doi.org/10.1109/MSP.2017.71

Kiayias, A., Zacharias, T., & Zhang, B. (2017b). Ceremonies for end-to-end verifiable elections. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/978-3-662-54388-7_11

Knezevic, M., Nikov, V., & Rombouts, P. (2016). Low-Latency ECDSA Signature Verification-A Road Toward Safer Traffic. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. https://doi.org/10.1109/TVLSI.2016.2557965

López-García, L., Dominguez Perez, L. J., & Rodríguez-Henríquez, F. (2014). A pairing-based blind signature E-voting scheme. Computer Journal. https://doi.org/10.1093/comjnl/bxt069

Mateu, V., Miret, J. M., & Sebé, F. (2016). A hybrid approach to vector-based homomorphic tallying remote voting. International Journal of Information Security. https://doi.org/10.1007/s10207-015-0279-8

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System | Satoshi Nakamoto Institute. In 2008-10-31.

Panja, S., & Roy, B. K. (2018). A secure end-to-end verifiable e-voting system using zero knowledge based blockchain. IACR Cryptology E-Print Archive.

Quaglia, E. A., & Smyth, B. (2018). Secret, verifiable auctions from elections. Theoretical Computer Science, 730, 44–92. https://doi.org/10.1016/j.tcs.2018.03.022

Sandler, D., Derr, K., & Wallach, D. S. (2008). VoteBox: a tamper-evident, verifiable electronic voting system. Proceedings of the 17th Conference on Security Symposium.

Tarasov, P., & Tewari, H. (2017). Internet voting using Zcash. Proceedings of the International Conference on WWW/Internet 2017 and Applied Computing 2017.

Vijayalakshmi, S., & Karpagam, G. R. (2018). Secure online voting system in cloud. Electronic Government. https://doi.org/10.1504/EG.2018.093407

Xiang, G., Yu, B., & Zhu, P. (2012). A algorithm of fully homomorphic encryption. Proceedings - 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2012. https://doi.org/10.1109/FSKD.2012.6234023

Yin, H. L., Fu, Y., & Chen, Z. B. (2016). Practical quantum digital signature. Physical Review A. https://doi.org/10.1103/PhysRevA.93.032316

Yu, B., Liu, J. K., Sakzad, A., Nepal, S., Steinfeld, R., Rimba, P., & Au, M. H. (2018). Platform-Independent Secure Blockchain-Based Voting System. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/978-3-319-99136-8_20