IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-12, Dec 2019 E-ISSN: 2347-2693

# Homomorphic Encryption: Privacy Preserving Amicable E-voting System

**Bhumika Patel[1*], Dharmendra Bhatti[2]**

[1]Babu Madhav Institute of Information Technology, Uka Tarsadia University, Bardoli, India
[2]Shrimad Rajchandra Institute of Management and Computer Application, UTU, Bardoli, India

**Corresponding Author: bhumika.patel@utu.ac.in*

DOI: https://doi.org/10.26438/ijcse/v7i12.4650 | Available online at: www.ijcseonline.org

Accepted: 10/Dec/2019, Published: 31/Dec/2019

***Abstract*** **-**Advancement in technology plays a vital role in adherence of democratic processes. While making technology encroachment, democratic nation smoothens to the process of e-voting for civilians. Compromising security is an adverse effect in progression of online easy going processes. Trust and privacy are at risk especially in online vote storage. One way to protect stored data is to apply encryption with the condition that only recipient can decode those data. This technique can be carried out with online electronic voting system to prevent vote tampering from insider or outsider adversaries.
This research has been carried out to achieve privacy preservation and increasing trust factor among voters. To achieve given objective various additive Homomorphic encryption algorithms are implemented and as a result proved that paillier's Homomorphic encryption is the effective algorithm to be implemented to accomplish privacy on casted votes.

***Keywords*** – Homomorphic encryption, e-voting, paillier

## I. INTRODUCTION

General Elections are carried out in every democratic country between regular interval of years so as to elect a public- favoured administration and ruling party for the nation. These democratic nations use ballot voting system where- in the eligible citizens of the nation within a specific range of age cast their vote for their favourable candidate from a list of participating candidates, competing for the election. These ballot-voting systems which in general are conducted through Electronic Voting Machines have been used since years in different nations. With the advancement of technology, the traditional EVM systems is gradually becoming obsolute and replaced by electronic EVM systems wherein the whole voting process ia managed electronically through a centralised network. However, the introduction of networking technologies have paved the way in for a large number of security loopholes and vulnerabilites especially data privacy which leads to compromising of the whole voting process thus resulting into loss of integrity in casted votes , manipulation in votes and even loss of votes. Hence it is now mandatory to improvise the security standards of the electronic voting systems looking to the advancement of technology and future requirements.

The traditional standard security procedure to defend against security threats is to use encryption. Though there are advance measures available too be used, encryption remains the most preferred, robust and unbreachable approach. In general, encryption can be done through symmetric and asymmetric methods.[1] But, most of the encryption methods have their own disadvantages such as calculation overhead, time overhead, process complexity and lack of space to store original as well as encrypted data. Also, techniques like bruteforce, man-in-the-middle attacks and traffic analysis could be used by attackers to break the encyrption or in some cases even bypass it. Apart from this, using encryption makes it compulsory to decrypt the original data if a certain kind of processing or calculation over the data is required to be carried out. This would create a major priavcy concern in an Electronic voting system as a decrypted casted vote could be easily interpreted by the attacker. The homomorphic encryption facilitates the user with the ability to process over encrypted data without having to decrypt it.[1] Thus, homomorphic encryption can be a probable solution for an Electronic voting system.

We have proposed secured framework for cloud based E-voting [2]. One component of that framework was applying homomorphic encryption. In the present study, we described the component along with novel secure model, method and implementation. This model aims to address the issue of privacy preservation in the voting process through Electronic voting system using homomorphic encryption. The implemented model has also been tested for the effectiveness of security by using different algorithms of additive homomorphic encryption and then comparing results of casted vote. The results show that the proposed model is successful in providing privacy preservation to Electronic voting system.

 

## II. BACKGROUND STUDY

Homomorphism allows computations to be carried out on ciphertext. Cipher text need not to be decrypted before totalling of vote, thus achieves privacy on casted votes. Basically two types of homomorphic encryptions are available : Additive homomorphic and multiplicative homomorphic encryption. Additive homomorphic encryption is required as the work has been carried out on e-voting where tallying of vote is to be done as last phase of election.

Various homomorphic encryptions are available in the category of additive homomorphism. Goldwasser and Micali has invented first homomorphic encryption in the year of 1982[3]. The method converts the message in to stream of bits and each bit is encrypted seprately which leads to ciphertext expansion in large amount. This method is best suited in encrypting binary numbers. As an advancement of this method Benaloh has been invented as second homomorphic encryption. This is the first homomorphic encryption which encrypts whole block instead of bit by bit.[4] Naccache-N-Stern has been invented as a generalization of Benaloh. Decryption process has been changed to increase the computation efficiency in Benaloh.[5] Next advancement in the homomorphism is Elgamal. This method is advanced version of Diffie Hellman Key exchange algorithm.[6] Though it is multiplicative in nature but if range of generators are fixed then it can behave as additive homomorphic algorithm. After all Paillier encryption method has been invented by Pascal Paillier in the year of 1999.[7] Paillier is the improved version of all the above stated methods of homomorphic encryption. In comparision of all, paillier decreases the ciphertext expansion. It uses the famous chinese reminder therom to sturdy reduction in decryption.[7] Paillier results into smaller expansion and cost reduction in compare to all other schemes.[8] Algorithm is enriched with various properties like, multiplying encrypted messages results into addition of all plaintext and ciphertext can be changed without changing original plaintext.[9] Paillier is the outstanding option to achieve privacy using homomorphism.[10]

## III. RESEARCH METHODOLOGY AND IMPLEMENTATION

### A. Choosing the Appropriate Encyrption Algotihm

As specified in the introduction , most of the encryption methods have their own disadvantages such as calculation overhead, time overhead, process complexity and lack of space to store original as well as encrypted data.Also, in general any general election includes a million number of citizens who shall be casting votes, thus leading to an enormous amount of data to be stored and processed by the e-voting system. Due to this it is very important to implement encryption in a way that overcomes all this issues. Hence, the researcher has implemented different methods of encryption by using 100 random numbers and after proper analysis and result comparison, pallier algortihm has been chosen to be the optimal solution to be used for encyption.

### B. Process of comparision of encryption algorithms

The comparision of algorithms was on Goldwasser-Micali, Benaloh, Naccache-N-Stern, Elgamal and Paillier. For comparing the encryption algortihms, 100 random numbers were encrypted by each algorithm. A CSV file was made which contained key initialization time, encryption time, decryption time, total time taken and total size taken. Average for each parameter was calculated and then compared for accuracy.

Table 1: Homomorphic Algorithm Comparision

| Algorithm Name | Key Init Time | Encyption Time | Decryption Time | Total Time | Total Size Taken |
|---|---|---|---|---|---|
| Goldwasser-Micali | 0.4799841 | 0.0001147 | 0.0000782 | 0.4801770 | 371.59 |
| Benaloh | 1.7214342 | 0.0001516 | 0.0001238 | 1.7217095 | 81.68 |
| Naccache-N-Stern | 0.0546679 | 0.0722889 | 0.0125067 | 0.1394635 | 183.44 |
| Elgamal | 0.0000274 | 0.0000073 | 0.0000051 | 0.0000397 | 92.00 |
| Paillier | 0.0561540 | 0.0157827 | 0.0207673 | 0.0927041 | 32.00 |

By considering parameteres and obtain results researcher has arrived with the conclusion that Paillier gives the efficient result among all the algorithms. Below given graph shows the total time and size consumed by paillier is less among all algorithms.

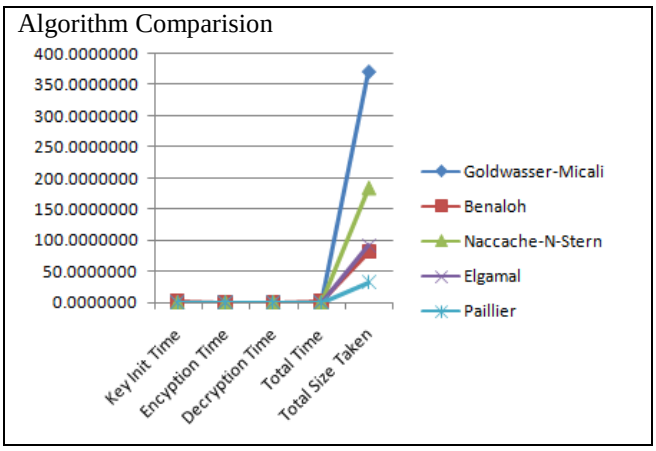


Figure 1: Homomorphic Algorithm Comparirion

### C. Proposed Model

The proposed model is composed of a client, server and an dministrator.

The process of voting by using homorphic encryption in the proposed model is as follows:

 

1. Voting
2. Encryption
3. Storage
4. Tallying
5. Decryption
6. Result

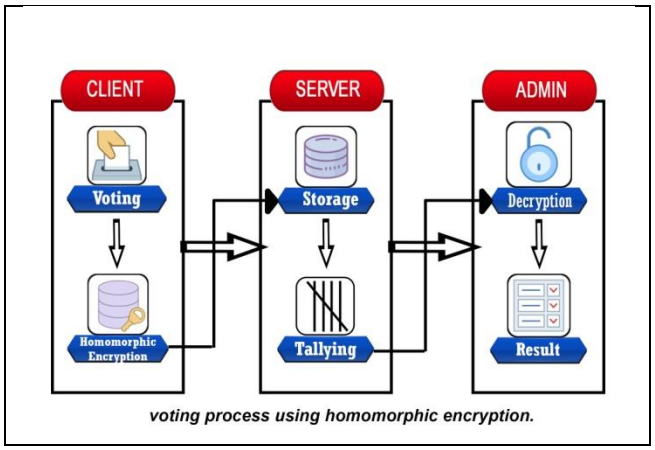


Figure 2. System Design

The process initiates with the user casting a vote to his favourable candidate, by selecting from the user menu visible to him upon authentication. The user casted vote is then encrypted using additive pallier encryption on the client side and sent to the server for being stored. After all the votes have been casted, on the server side, a tally process is done. This tally process counts all the votes of the user and also seperates the total vote count for all the candidates, thus providing a final count of candidate-wise votes for each candidate. From this process the candidate having maximum candidate-wise count of votes is identified. This information is again stored on the server after encryption. Finally, the administrator clicks on result to see the winner, where in backend the result is decrypted and displayed to the admisnitrator.

### D. Paillier Encryption Algorithm

Key Generation

Paillier algorithm has two keys: public key and private key for encrypting and decrypting messages. And the process of generating keys is as follows: [9]

1. Choose two large prime numbers p and q randomly and independently of each other such that gcd(pq,(p−1)(q−1))=1
2. Compute n=p.q
3. Compute λ=lcm(p-1,q-1)
4. Select generator g where g∈ $Z^*n^2$
5. Calculate modular multiplicative inverse
   μ= (L(gλ mod n2))-1 mod n
   Where function L is defined as L(x) =x-1/n
   Public key (encryption key) is (n, g)
   Private Key (decryption key) is (λ, μ)

Encryption

1. Let m be a message to be encrypted where m ∈ Zn
2. Select random r where r ∈ Z*n
3. Compute cipher text as: C= gm*rn mod n2

Tallying

$T=\prod_{i=1}^{Nv} Ci \bmod n2$

Decryption

Compute message: $m= (c^{\lambda} \bmod n^2) \cdot \mu \bmod n$

### E. Methodology Implementation

As part of simulation the designed e-voting scheme has been implemented in Python 3.7, c# .Net and SQlite database. For efficient implementation of this conception each registered candidate of specific constituency will be allocated specific number as a vote. Upon selection of candidate by voter the allocated number will be considered as a vote for further encryption during voting. Below table shows the process.

Example

1. So we choose primes randomly $p$ = 293, $q$ = 433
2. gcd(pq,$(p-1)(q-1)$)=1 holds here
3. n= pq = 126869 $n^2$= 16095743161
4. λ= lcm(p-1, q-1) = 31536
5. We choose generator g where g∈ $Z^*n^2$ = 6497955158
6. μ = $((6497955158^{31536} \bmod 16095743161) - 1 / 126869)^{-1} \bmod 126869$

Table 2: Original Votes

| Voter Name | Bhavin Patel $10^0$ | Mihir Vyas $10^1$ | Rasesh Dave $10^2$ | Dhaval Solanki $10^3$ | Amit Vyas $10^4$ | Viral Ahir $10^5$ | Vote Message |
|---|---|---|---|---|---|---|---|
| Vyana | ✓ | | | | | | $10^0$ |
| Sejal | | ✓ | | | | | $10^1$ |
| Stuti | | | ✓ | | | | $10^2$ |
| Mitali | | | | ✓ | | | $10^3$ |
| Jalpa | | | | | ✓ | | $10^4$ |
| Neha | | | | | | ✓ | $10^5$ |
| Hetal | ✓ | | | | | | $10^0$ |
| Ramesh | ✓ | | | | | | $10^0$ |
| Sangita | ✓ | | | | | | $10^0$ |
| Piyush | | ✓ | | | | | $10^1$ |

## IV. RESULTS AND DISCUSSION

Several experiments has been conducted to prove effectiveness of paillier cryptosystem and homomorphism. The conducted experiments are algorithm implementation test, ciphertext distinctiveness, homomorphic test and message expansion test.

### A. Algorithm Implementation Experiment

In this experiment voting has been done by 10 voters in implemented model. This test has been carried out to check correctness of the algorithm. The obtain result is shown in voting result table.

 

Table 3: Vote Encryption

| Voter Name | Vote | Random number | Encrypted value |
|---|---|---|---|
| Vyana | $10^0$ | 35145 | 5317389514 |
| Sejal | $10^1$ | 74384 | 13344258618 |
| Stuti | $10^2$ | 10966 | 8480290938 |
| Mitali | $10^3$ | 17953 | 3871259671 |
| Jalpa | $10^4$ | 7292 | 14899747940 |
| Neha | $10^5$ | 24819 | 4625828449 |
| Hetal | $10^0$ | 4955 | 5842821276 |
| Ramesh | $10^0$ | 118037 | 3033281324 |
| Sangita | $10^0$ | 96584 | 15738470018 |
| Piyush | $10^1$ | 10966 | 6755046122 |

Homomorphic Tallying

T=3025175311668293982282704419053657034632973631707273268793091813434766360418218101841204062952 55040

T=11854008514

Decryption

Compute message: $m = (c^{\lambda} \bmod n^2) \cdot \mu \bmod n$

m= 111124

Result

$111124 = 1*10^5+1*10^4+1*10^3+1*10^2+2*10^1+4*10^0$

Table 4: Voting Result

| Candidate Name | Vote | Vote count | Total Vote |
|---|---|---|---|
| Bhavin Patel | $10^0$ | $4*10^0$ | 4 |
| Mihir Vyas | $10^1$ | $2*10^1$ | 2 |
| Rasesh Dave | $10^2$ | $1*10^2$ | 1 |
| Dhaval Solanki | $10^3$ | $1*10^3$ | 1 |
| Amit Vyas | $10^4$ | $1*10^4$ | 1 |
| Viral Ahir | $10^5$ | $1*10^5$ | 1 |

### B. Ciphertext distinctiveness Experiment

This test demonstrates that the rendom value r generates distinct ciphertext for same plaintext.

Table 5: Ciphertext Distinctiveness

| Vote | Random number | Encrypted value |
|---|---|---|
| $10^0$ | 35145 | 5317389514 |
| $10^0$ | 4955 | 5842821276 |
| $10^0$ | 118037 | 3033281324 |
| $10^0$ | 96584 | 15738470018 |

### C. Homomorphic Experiment

Table 6: Homomoephism verification

| No | M1+M2 | Cipher text | Decryption | Conclusion |
|---|---|---|---|---|
| 1 | 15+25 | 761192151692599036 | 40 | True |
| 2 | 1517+17 | 3285374471141930478 | 1534 | True |
| 3 | 2500+2600 | 10725175570749887197 | 5100 | True |
| 4 | 5+9 | 1724256700564626676 | 14 | True |

This experiment ensures the homomorphic property of paillier algorithm mentioned in background study section that individual ciphertext need not to be decrypted for getting result.

### D. Message Expansion Experiment

Increment in key size always generates strong ciphertext. Algorithm has been tested over 32, 64, 128, 256 and 512 key size. And it shows the expansion in message to strengthen the encryption process.

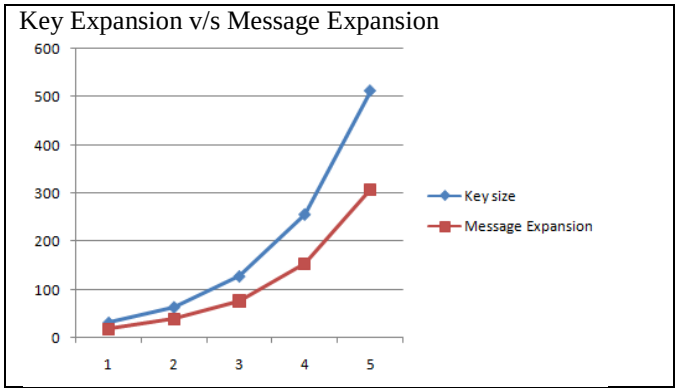


Figure 3: Message Expansion

By considering all the experiments done on the model it can be declared that the proposed algorithm performs as expected and generate all correct results.

## V. CONCLUSION AND FUTURE WORK

This research demonstrates the privacy preserving homomorphic encryption enabled electronic voting system. Inoredr to achieve privacy, paillier is the best option to encrypt ballot. Through out voting process casted ballots remains confidential due to homomorphism which doesn't allow single vote to be decrypted for firther calculation. As well as due to exquisiteness of rendom number used in encryption process it generates completely unique ciphertext for each voter.

Confidentiality, integrity and availability are the three pillers of the security. Further research focuses on the approach to achieve integrity in e-voting process.

## REFERENCES

[1] A. Saranyadevi, S. Anguraj, S. Senbhaga, "*A Detailed Study on Homomorphic Encryption*", International Journal of Morden Trends in Engineering and Research, ISSN: 2349-9745

[2] B. Patel, D. Bhatti, "*A Proposed Secured Framework for Cloud Based E-Voting*" in the proceeding of International Conference on New Frontiers of Engineering, Science, Management and Humanities (ICNFESMH-2018), pp. 364-369, 2018.

[3] Goldwasser, S. & Micali, S. "*Probabilistic Encryption and How to Play Mental Poker Keeping Secret All Partial Information*." 14th

 

Annual ACM Symposium on Theory of Computing (STOC'82), pp. 365-377, 1982.
[4] Benaloh, J. , "*Verifiable Secret-Ballot Elections. Doctoral Dissertation*", Department of Computer Science, Yale University, New Haven, Connecticut, USA., 1988
[5] Naccache, D. & Stern, J, "*A New Public Key Cryptosystem Based on Higher Residues*." 5th ACM Conference on Computer and Communications Security (CCS'98), pp. 59-66, ACM Press, New York, NY, USA., 1998
[6] Maha TEBAA, Saïd EL HAJJI, Abdellatif EL GHAZI, "Homomorphic Encryption Applied to the Cloud Computing Security", Proceedings of the World Congress on Engineering 2012 Vol I WCE 2012, July 4 - 6, 2012, London, U.K.
[7] Paillier, P, "*Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In: Advances in Cryptology*" – Proceedings of EUROCRYPT'99, Lecture Notes in Computer Science (LNCS), Vol 1592, Springer-Verlag, pp. 223-238. 1999
[8] Jaydeep Sen, "*Homomorphic Encryption:Theory and Application*," NIT Odisha
[9] Michael O'Keeffe, "*The Paillier Cryptosystem – A look Into The Cryptosystem And Its Potential Application*", http://www.tcnj.edu/~hagedorn/papers/CapstonePapers/OKeeffe/CapstoneOKeeffeCryptography.pdf.
[10] S. Sinde, S. Shukla, D.K.Chitre,"*Secure E-voting Using Homomorphic Technology*", International Journal of Emerging Technology And Advanced Engineering, Volume 3, Issue 8, ISSN : 2250 – 2459

**Author's Profile**

Prof. Dr. Dharmendra G. Bhatti Professor & Information Technology Head, Uka Tarsadia University, Bardoli-394350, Dist. Surat, Gujarat, India. He has completed his Ph. D. in Computer Science and having 17 research publications.
Presently guiding 7 Ph. D. (Computer Science) students. As Information Technology head, significantly contributed in, in-house University ERP development and transformed major university processes from manual/paper based to online/digital.

Ms. Bhumika Patel Assistant Professor & Training & Placement Officer, Babu Madhav Institute of Information Technology, Uka Tarsadia University, Bardoli-394350, Dist. Surat, Gujarat, India. She has completed her MCA from Gujarat University. She is currently pursuing her Ph. D in Computer Science and having 5 research publications.

 