BÁO CÁO ĐỒ ÁN CUỐI KỲ

HOMOMORPHIC ENCRYPTION – BASED ONLINE VOTING SYSTEM

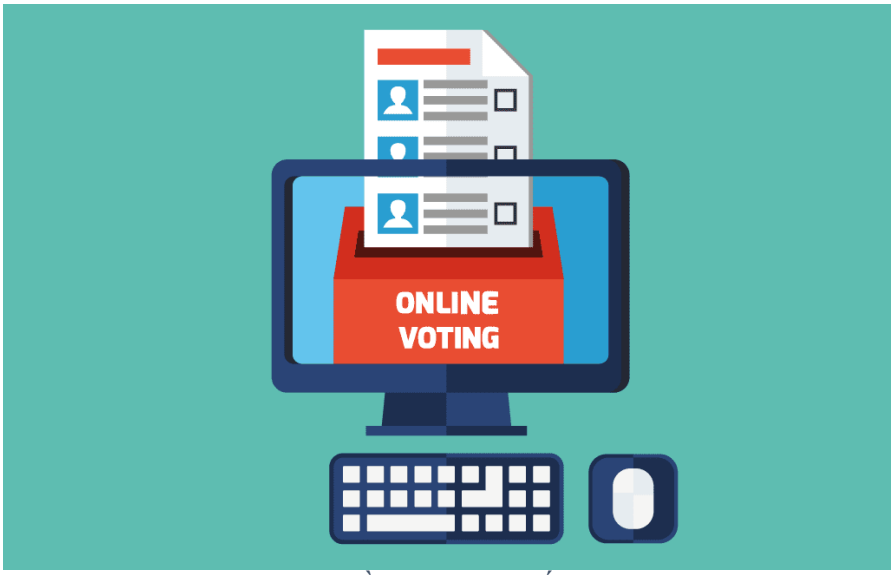
NT219.L21.ATTN
GVHD: Nguyễn Ngọc Tự

Trần Đức Lương - 19521815
Vương Minh Tiến - 19522346

MỤC LỤC

I. Context

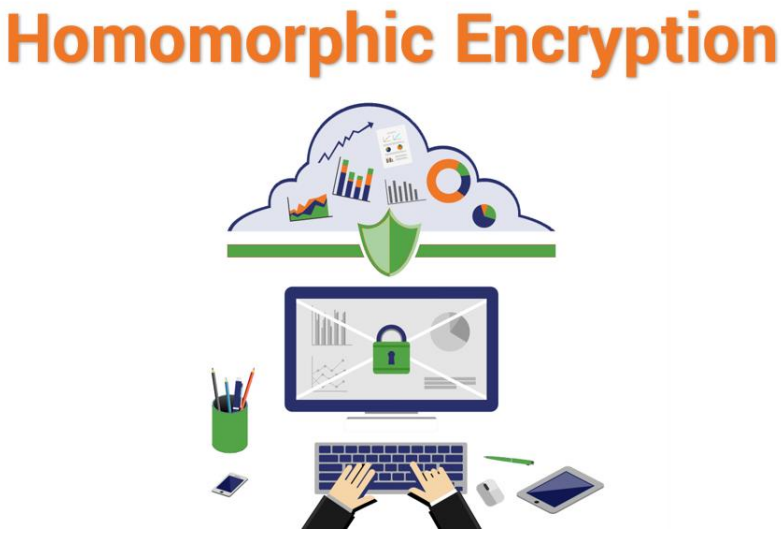
Bầu cử là một quá trình tổng hợp phiếu bầu của người dân thông qua bỏ phiếu để chọn ra một cá nhân nắm giữ các chức vụ trong nền dân chủ. Tuy nhiên trong bỏ phiếu truyền thống, mỗi cử tri phải đi tới điểm bỏ phiếu và bỏ phiếu. Điều này gây khó khăn trong khâu tổ chức, cũng như khá tốn thời gian cho tất cả mọi người. Bên cạnh đó, những người ở vùng sâu vùng xa hoặc công tác xa nhà cũng sẽ gặp bất tiện lớn. Với sự phát triển gần đây của công nghệ, bỏ phiếu trực tuyến ra đời giúp khắc phục những hạn chế trên, những cử tri có thể bầu cho các ứng cử viên ở bất cứ đâu thông qua Internet. Việc bầu cử trở nên rất thuận lợi, chỉ thông qua một số thao tác đơn giản trên thiết bị di động hay máy tính mà không ảnh hưởng đến công việc, hay phải xếp hàng đợi như cách bầu cử truyền thống. Tuy nhiên, kết quả được gửi đi qua internet ở dạng thô (raw) rất có thể bị lộ ra ngoài và dễ dàng chỉnh sửa trong quá trình vận chuyển. Với việc sử dụng mã hóa dữ liệu trước khi gửi đi, vấn đề trên được đảm bảo nhưng sẽ làm phát sinh vấn đề khác là dữ liệu vẫn sẽ bị lộ khi hệ thống cần giải mã để thực hiện tính toán phiếu bầu. Trong đề tài này, nhóm em đề xuất giải pháp để bảo mật dữ liệu cử tri gửi đi bằng phương pháp mã hóa và đồng thời đảm bảo việc tính toán kết quả bầu cử chỉ dựa trên bản mã.



Hình 1: Bầu cử trực tuyến [1] [2]

II. Research Motivations

Như đã đề cập ở phần ngữ cảnh, việc tổ chức bầu cử trên mạng vẫn tồn tại mặt trái trong vấn đề bảo mật thông tin [1] [2]. Cụ thể, khi cử tri gửi dữ liệu đi qua Internet, họ không thể quản lý và đảm bảo được tính toàn vẹn và riêng tư của phiếu bầu. Server hoàn toàn kiểm soát được thông tin lá phiếu gốc của từng cử tri. Những kẻ tấn công hoặc tổ chức quán lý phiếu bầu không đáng tin cậy có thể sử dụng và sửa đổi, làm mất tính công bằng, minh bạch của cuộc bầu cử. Hơn nữa, mỗi cử tri có thể lợi dụng và thực hiện quyền bầu chọn nhiều lần, gây nên tính sai lệch trong cuộc bầu cử. Đã có một số giải pháp để giảm thiểu những vấn đề trên, ví dụ như mỗi cử tri đều mã hoá phiếu bầu rồi gửi lên server, họ sẽ cung cấp khóa dùng cho việc giải mã để tổng hợp phiếu bầu. Nhưng với số lượng lớn cử tri tham gia bầu cử, việc mã hoá và giải mã để tính toán thì tốn rất nhiều thời gian, tài nguyên và không thể đảm bảo việc che giấu thông tin lá phiếu(plaintext) khỏi server. Vì vậy, chúng ta cần tìm ra một cơ chế thực hiện để giải quyết vấn đề vừa có thể che giấu bản rõ với server mà vừa có thể tính toán để tổng hợp phiếu bầu cho cuộc bầu cử kín [1] [3]. Và Homomorphic Encryption là một phương thức mã hóa đáp ứng được điều đó.



Hình 2: Homomorphic Encryption [1] [3]

III. Proposed Scheme

1. *Homomorphic Encryption*

Trong khi đó, Homomorphic Encryption (HE) [1] [3] đang được giới nghiên cứu quan tâm nhiều trong những năm trở lại đây. HE là mô hình mã hóa đồng cấu (nghĩa là, khi nhân hai bản mã bất kỳ thì khi giải mã kết quả sẽ nhận được bản rõ là tích của hai bản rõ tương ứng ban đầu) cho phép chúng ta làm việc, tính toán (phép cộng và phép nhân) dựa trên bản mã tương tự như làm việc trên bản rõ. Điều này giúp không chỉ tiết kiệm thời gian mà còn đảm bảo tính bảo mật cao cho dữ liệu của người dùng. Có 3 dạng Homomorphic Encryption chính được sử dụng hiện nay và chúng được phân biệt dựa trên đặc điểm và tần suất các phép toán có thể thực hiện được dựa trên ciphertext. Với Partially Homomorphic Encryption cho phép tính toán một phép tính (cộng, nhân) nhưng không giới hạn số lần thực hiện. Somewhat Homomorphic Encryption có thể thực hiện hai phép tính cùng nhau nhưng bị giới hạn số lần thực hiện dựa trên cài đặt. Cuối cùng là Fully Homomorphic Encryption hỗ trợ tính toán 2 phép tính trên số lần không giới hạn.

2. *Paillier Algorithm*

Dựa trên ý tưởng của RSA và El-Gamal, Pascal Paillier đã cho ra mắt Pailler Cryptosystem vào năm 1999 [4]. Đây là hệ thống dựa trên hệ thống mật mã bất đối xứng(Public-key Cryptosystem).Paillier Cryptosystem chính là Partially Homomorphic Encryption mã hóa đồng cấu trên phép cộng dựa trên dữ liệu đã được mã hoá [2] [5]. Paillier-cryptosystem được định nghĩa dựa trên 4 thành phần: tạo khoá (Key Generation), mã hoá (Encryption), giải mã (Decryption) và tính toán (Homomorphic Addition).

Key Generation:

- $p, q \in P$ with equal length
- $n = pq$
- $g = 1 + n$
- $\phi(n) = (p-1) \cdot (q-1)$
- $\mu = \phi(n)^{-1} \bmod n$ (μ is used as *private-key*)

Encryption:

- Plaintext $m < n$ $(m \in Z_n)$
- Choose $r < n$ dengan $\gcd(r, n) = 1$ randomly $(r \in Z_n^*)$
- Ciphertext $c = g^m . r^n \bmod n^2$

Decryption:

- Ciphertext $c < n^2$
- Plaintext $m = L(c^\lambda \bmod n^2) \cdot \mu \bmod n$

Hình 3: Công thức của Paillier Cryptosystem [4]

Tương tự với các loại mã hóa bất đối xứng như RSA, hệ thống khóa của Paillier Cryptosystem [4] cũng bao gồm public key (n, g) để mã hóa và private key (λ, μ) để giải mã, trong đó modulus n được cấu thành bằng tích của hai số nguyên tố random đủ lớn p và q. Khi đó với mỗi giá trị r random trong khoảng $(0, n)$ sao cho $gcd(r, n) = 1$, cùng với một bản rõ m sẽ cho ra các bản mã c khác nhau theo công thức $c = g^m r^n \ mod \ n^2$. Đó chính là vai trò của giá trị r, nhằm tăng tính bảo mật (security) và riêng tư (privacy) của dữ liệu. Quá trình giải mã cũng được thực hiện dễ dàng với công thức $m = L(c^\lambda \ mod \ n^2) \ \mu \ mod \ n$, trong đó hàm L được định nghĩa với $L(x) = \frac{x-1}{n}$.

Từ công thức mã hóa trên, chúng ta có thể thấy được thuật toán Paillier là một dạng Additive Partially Homomorphic Encryption, nghĩa là với public key (n, g) và hai bản mã của m_1 và m_2 ta có thể tính toán được bản mã của $m_1 + m_2$ bằng cách lấy $c_1 \times c_2$ mà không cần phải giải mã. Ta sẽ chứng minh tính chất trên như sau:

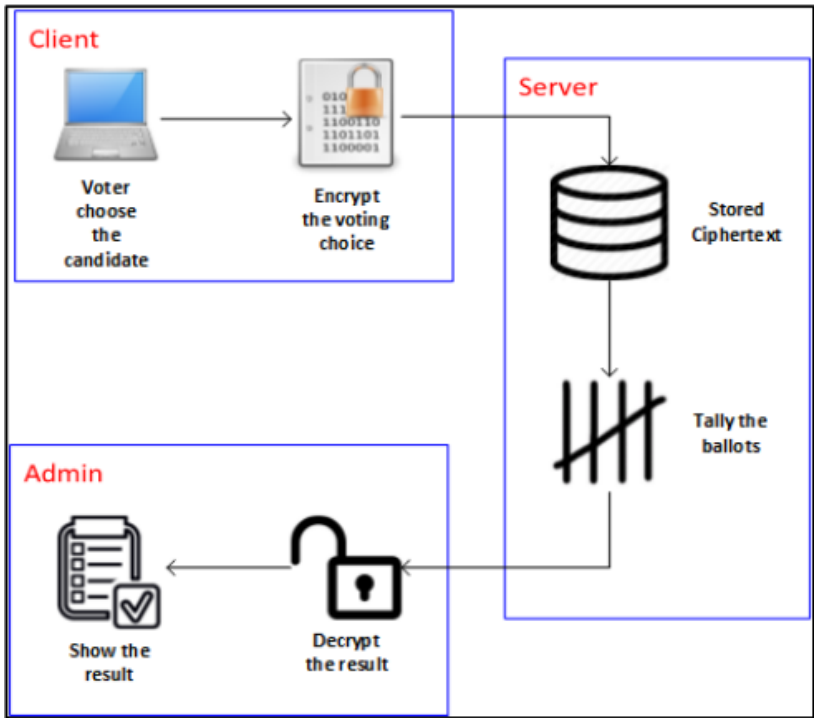
$$E(m_1) = c_1 = g^{m_1} {r_1}^n \ mod \ n^2 \quad (1)$$

$$E(m_2) = c_2 = g^{m_2} {r_2}^n \ mod \ n^2 \quad (2)$$

$$\xRightarrow[(1)(2)]{} c_1 \times c_2 = g^{m_1+m_2}(r_1 r_2)^n \ mod \ n^2 = E(m_1 + m_2)$$

Tính chất trên rất phù hợp để áp dụng trong việc tính toán tổng số phiếu bầu trong một cuộc bầu cử kín mà không cần phải biết giá trị bản rõ của từng phiếu bầu. Đây là lý do nhóm em sử dụng thuật toán Paillier cho đề tài này.

3. Voting Procedure



Hình 4: Online Voting Procedure using HE [2]

Dựa trên những phân tích trên, trong đồ án lần này, nhóm em xin đề xuất mô hình hệ thống bỏ phiếu trực tuyến hoạt động sử dụng Paillier Cryptosystem để tính toán phiếu bầu [1] [2]. Mỗi cử tri sử dụng thông tin do cơ quan bầu cử (Election Official) xác thực để truy cập hệ thống, sau đó thực hiện chọn những ứng viên muốn bầu cử. Tiếp theo, phiếu bầu của cử tri được gửi đi ở dạng mã hoá dựa trên public key do cơ quan bầu cử đáng tin cậy cung cấp, với giả thuyết rằng việc sharing key là an toàn và bảo mật. Cụ thể trong phiếu bầu, những ứng viên được bầu sẽ được gán bằng giá trị 1 (dạng plaintext), những ứng viên không được bầu sẽ được gán giá trị 0 (dạng plaintext). Những giá trị (0,1) trong phiếu bầu sẽ được mã hóa bằng Paillier sao cho các giá trị ciphertext tương ứng không trùng nhau. Dữ liệu ciphertext trên được gửi lên server và lưu trữ trong database. Khi đó, server chỉ có thể tính toán trên bản mã dựa vào tính chất additive của Paillier Cryptosystem để tổng hợp số phiếu mà không phân biệt được các cử tri đã bầu cho ai. Cách tính toán của server sẽ được trình bày như sau:

Đối với từng ứng cử viên, server sẽ lưu bản mã T là tổng số phiếu bầu dành cho ứng viên đó. Tại thời điểm cuộc bầu cử chưa bắt đầu, giá trị T được khởi tạo bằng bản mã Paillier của giá trị 0, gọi là c_0. Khi cuộc bầu cử bắt đầu, cứ sau mỗi lượt vote, server sẽ áp dụng tính chất additive của thuật toán Paillier để tính toán phiếu bầu tại thời điểm hiện tại của ứng viên đó. Cụ thể, quá trình tính toán được

chuyển thành công thức của dãy số, với T_i là số phiếu bầu dạng mã của ứng viên sau lượt vote thứ i :

$$\begin{cases} T_0 = c_0 \\ T_i = T_{i-1} \times c_i \ \ (mod\ n^2) \end{cases}$$

Cuối cùng sau khi quá trình bầu cử kết thúc, kết quả bầu cử T của từng ứng cử viên sẽ được gửi về cơ quan bầu cử để được giải mã dựa trên private key nhằm gửi kết quả về cho các cử tri. Quy trình bầu cử và xử lí kết quả đều được thực hiện trên bản mã thay vì bản rõ như cách bầu cử truyền thống.

IV. Contributions

Đồ án này nghiên cứu về mô hình hệ thống bỏ phiếu trực tuyến sử dụng thuật toán Paillier với thuộc tính additive HE [5] của nó. Mô hình đề xuất mang lại một giải pháp cực kì hiệu quả trong quá trình tổ chức các cuộc bầu cử so với cách tổ chức bầu cử truyền thống. Bằng việc mã hóa phiếu bầu gửi đi, dữ liệu của cử tri hoàn toàn được bảo mật, riêng tư thay vì bị toàn quyền kiểm soát bởi cơ quan bầu cử ở phương pháp truyền thống. Bên cạnh đó, khả năng tính toán kết quả phiếu bầu trên bản mã của thuật toán Paillier HE mang lại tính công bằng trong cuộc bầu cử kín, tránh những trường hợp cố tình sửa đổi kết quả ở bản rõ như trong cách tổ chức truyền thống. Đồng thời phương pháp được đề cập trong mô hình này mang lại hiệu suất cao bởi không gian key lớn (2048, 3072 bits), gấp rất nhiều lần so với số lượng cử tri bỏ phiếu trong một cuộc bầu cử.

V. Comment

Ứng dụng Homomorphic Encryption (HE) trong bầu cử trực tuyến có thể giúp bảo mật phiếu bầu trước khi gửi đi, đồng thời mang lại khả năng tổng hợp kết quả dựa trên bản mã; điều đó giúp tính trung thực, công bằng của cuộc bầu cử được đảm bảo khi trải qua các khâu trung gian. Tuy nhiên bầu cử trực tuyến cũng gặp phải một số khuyết điểm mà HE không thể xử lý được, chẳng hạn như việc xác thực một người dân đã từng tham gia bỏ phiếu hay chưa. Đây là vấn đề cần giải quyết vì nhiều cử tri có thể lợi dụng lỗ hổng này để thực hiện bỏ phiếu nhiều lần, gây nên tính sai lệch trong cuộc bầu cử.

Vậy nên với đề tài này, nhóm em xin đưa ra một hướng khắc phục sử dụng công nghệ Blockchain [6] để xác minh danh tính và quyền bầu cử của từng cử tri.

Trước tiên, thông tin cá nhân bao gồm cả chứng minh nhân dân (ID) của người dân sẽ được các tổ chức chính phủ cung cấp cho ủy ban bầu cử, cụ thể là danh sách những người dân đủ tuổi và đủ quyền công dân tham gia bầu cử. Ủy ban bầu cử sử dụng lượng dữ liệu này để cung cấp cho mỗi cử tri một ID token [6] dựa vào ID của người đó. Những ID token này là bằng chứng bầu cử của các cử tri, nó sẽ tồn tại nếu cử tri đó chưa tham gia bầu cử và sẽ bị xóa bỏ nếu người đó đã tham gia bỏ phiếu; Chúng được kí và xác thực tại ủy ban bầu cử. Sau đó, thông tin cá nhân kết hợp với các ID token sẽ được lưu vào một permissioned Blockchain của hệ thống bầu cử trực tuyến, có nghĩa là cử tri phải cần được phê duyệt để sử dụng Blockchain này.

Tiếp theo, quá trình bỏ phiếu được bắt đầu bằng việc đăng nhập xác minh của các cử tri. Để truy cập vào hệ thống, cử tri cần phải sử dụng họ tên và chứng minh nhân dân (ID) của mình. Sau khi hệ thống kiểm tra thông tin hợp lệ và có trong Blockchain, nếu như ID token còn tồn tại thì cử tri đó được thực hiện quyền bầu cử của mình một lần duy nhất. ID token của cử tri đó sẽ lập tức bị xóa bỏ khỏi hệ thống ngay sau khi quá trình gửi phiếu bầu hoàn tất. Khi đó, ở những lần đăng nhập tiếp theo, cử tri này chỉ có quyền xem những thông tin bên lề và kết quả của cuộc bầu cử mà không thể thực hiện quyền bỏ phiếu thêm một lần nữa.

Trên đây là đề xuất cơ bản của nhóm em trong việc hạn chế vấn đề bỏ phiếu nhiều lần (double-voting) nhằm tránh gian lận và mang lại tính công bằng cho cuộc bầu cử trực tuyến.

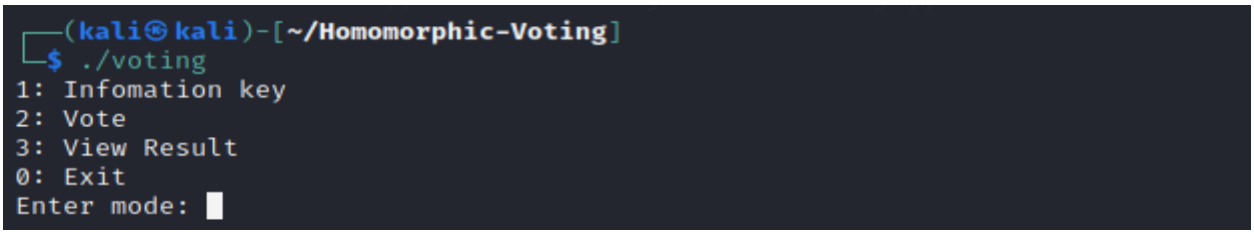
VI. Demo

Tại tài nguyên có hạn nên chương trình demo này không thực hiện trên mô hình client – server mà nhóm em chỉ mô phỏng lại trên máy tính cá nhân nhằm kiểm tra tính khả dụng của Homomorphic Encryption. Dưới đây là link github (bao gồm source code và cách biên dịch chương trình) của demo:

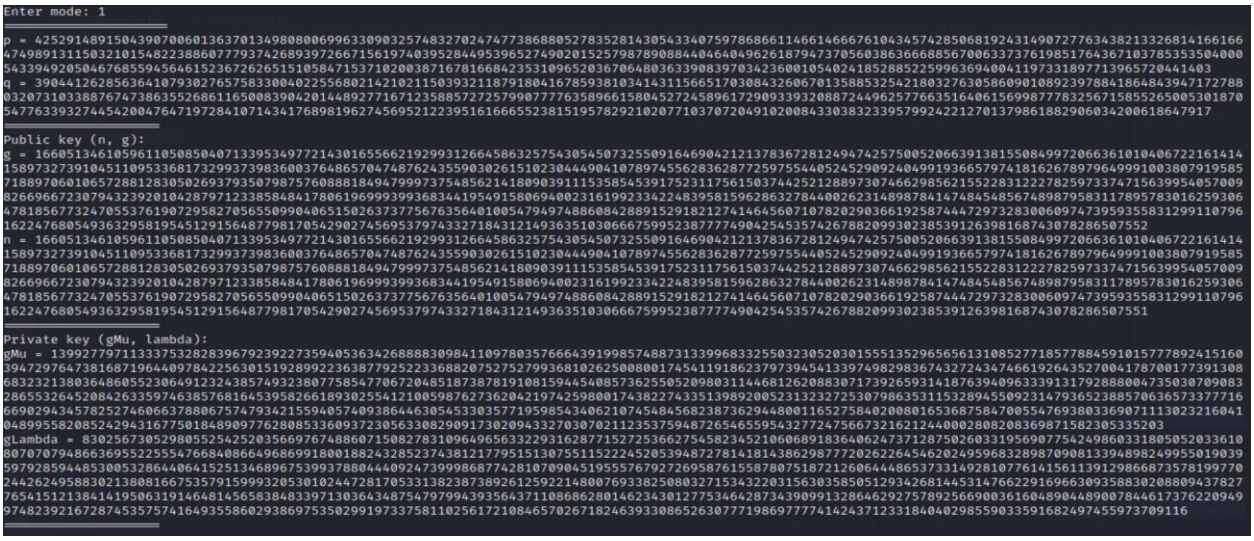
https://github.com/mt2651/Homomorphic-Voting.git

A. Main Demo

Chương trình có 2 hàm chính là ***Vote*** và ***View Result*** để thực hiện mã hóa, tổng hợp và giải mã, bên cạnh đó có hàm ***Information Key*** để mọi người có thể thấy rõ hơn được thông số của các key.



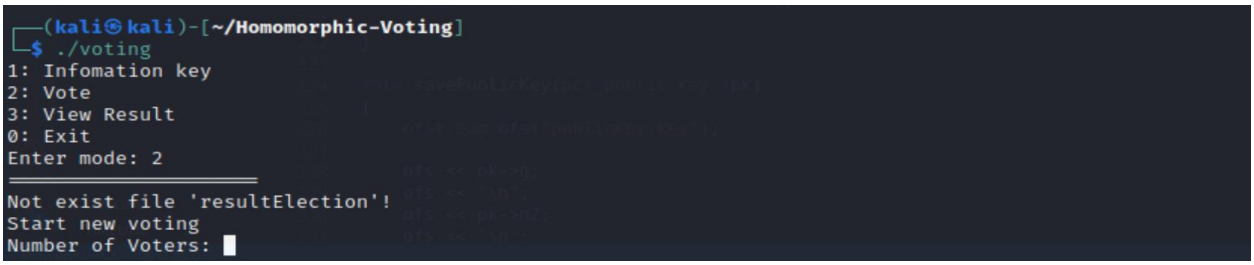
Hình 5: Giao diện chính



Hình 6: Thông số Key sử dụng

Đây là bộ key 3072 bits được tạo và lưu trong 2 file '**privateKey.key**' và '**publicKey.key**'.

Tiếp theo là hàm **Vote** với 2 quá trình là **Encryption** và **Addition 2 ciphertext** với số ứng cử viên candidate là 5, trong đó ciphertext thứ nhất là kết quả bầu cử dạng mã tính đến thời điểm bỏ phiếu được lưu ở file '**resultElection.txt**' và ciphertext thứ hai là phiếu bầu dạng mã của cử tri. Nếu file '**resultElection.txt**' không tồn tại, nghĩa là chưa có kết quả bỏ phiếu trước đó, ciphertext thứ nhất là các bản mã của giá trị 0 tương ứng cho tất cả các ứng viên.

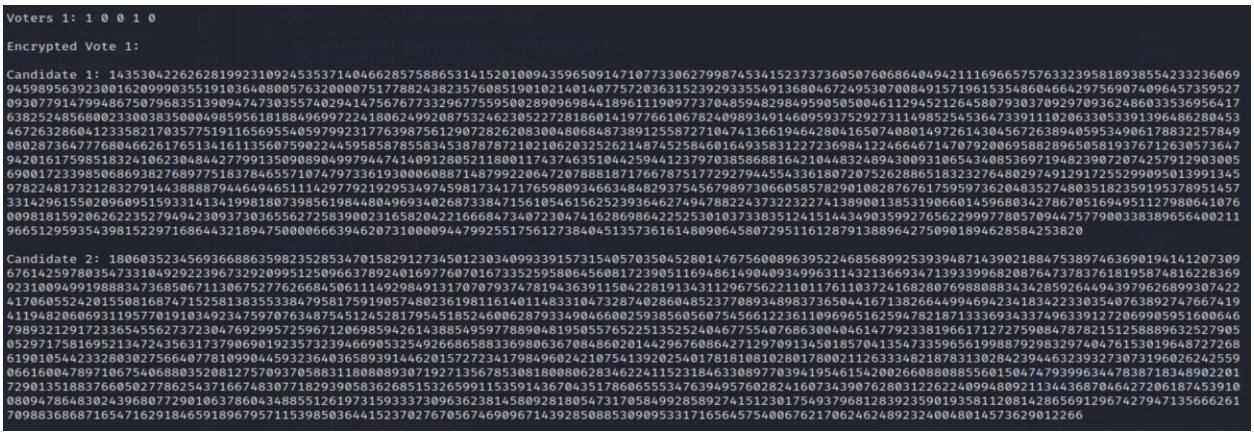


Hình 7: Bắt đầu bầu cử

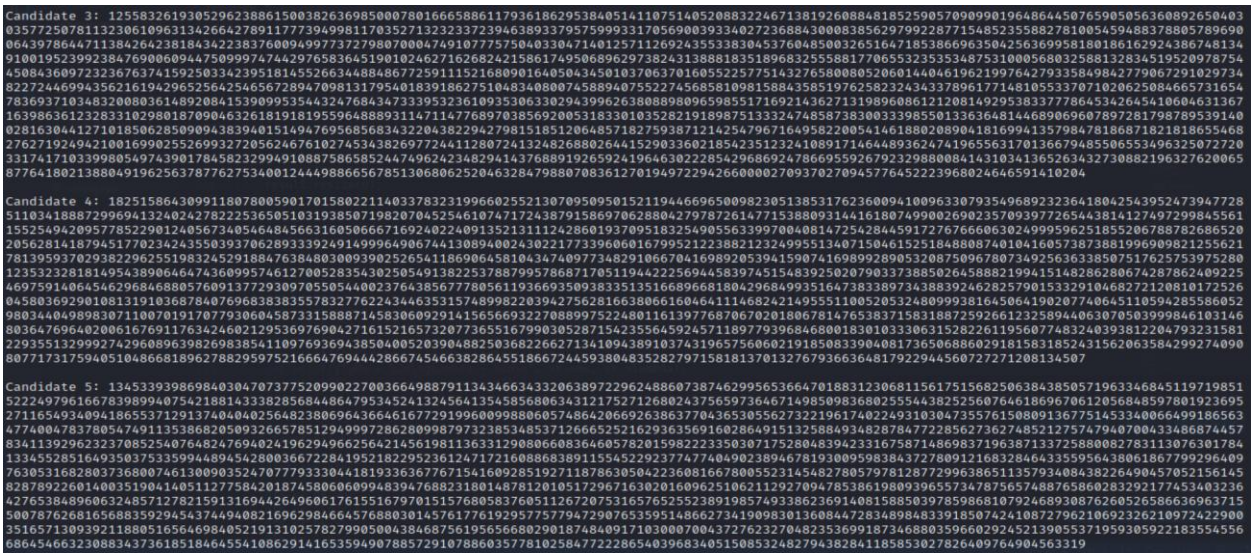
Đầu tiên, để có thể demo với nhiều cử tri, chương trình bắt nhập vào số cử tri tham gia bỏ phiếu, sau đó là kết quả bầu cử tương ứng với '1' là bầu, '0' là không bầu. Ở lần vote dưới, nhóm em điền số cử tri là 3.

Hình 8: Cử tri thực hiện bầu cử

Sau khi nhập vào kết quả, chương trình sẽ in ra phiếu bầu của cử tri sau khi được mã hóa.



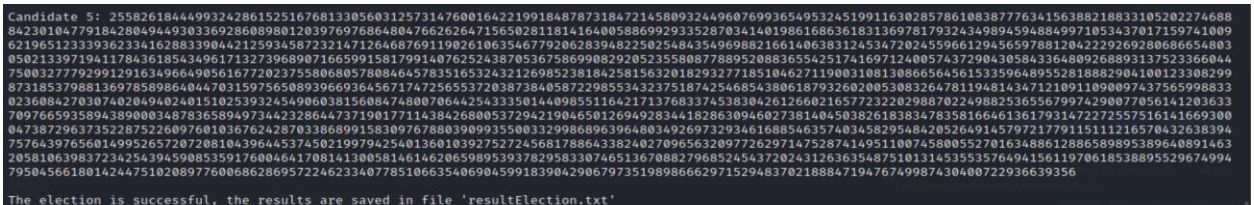
Hình 9: Phiếu bầu được mã hóa cho từng ứng cử viên(1)



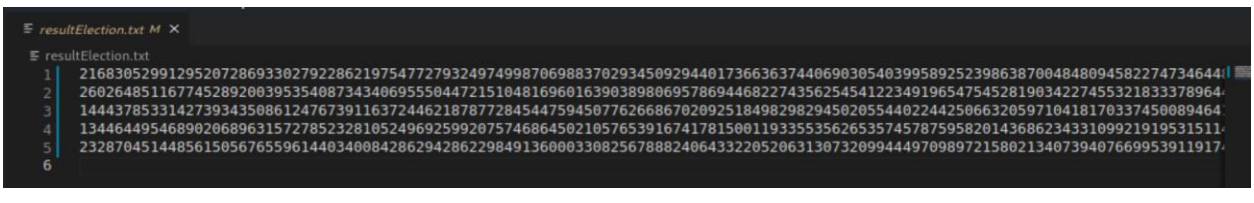
Hình 10: Phiếu bầu được mã hóa cho từng ứng cử viên(2)

Chúng ta có thể thấy cùng một plaintext là '0' hoặc '1' nhưng đều cho ra những ciphertext hoàn toàn khác nhau và không thể đoán được quy luật.

Tương tự cho 2 ứng cử viên còn lại, với mỗi lượt vote, chương trình sẽ tiếp tục **Encryption** và **Addition 2 ciphertext** đối với mỗi phiếu bầu của các ứng cử viên. Sau khi thực hiện các bước trên xong, kết quả cập nhật trở lại vào file '**resultElection.txt**'.



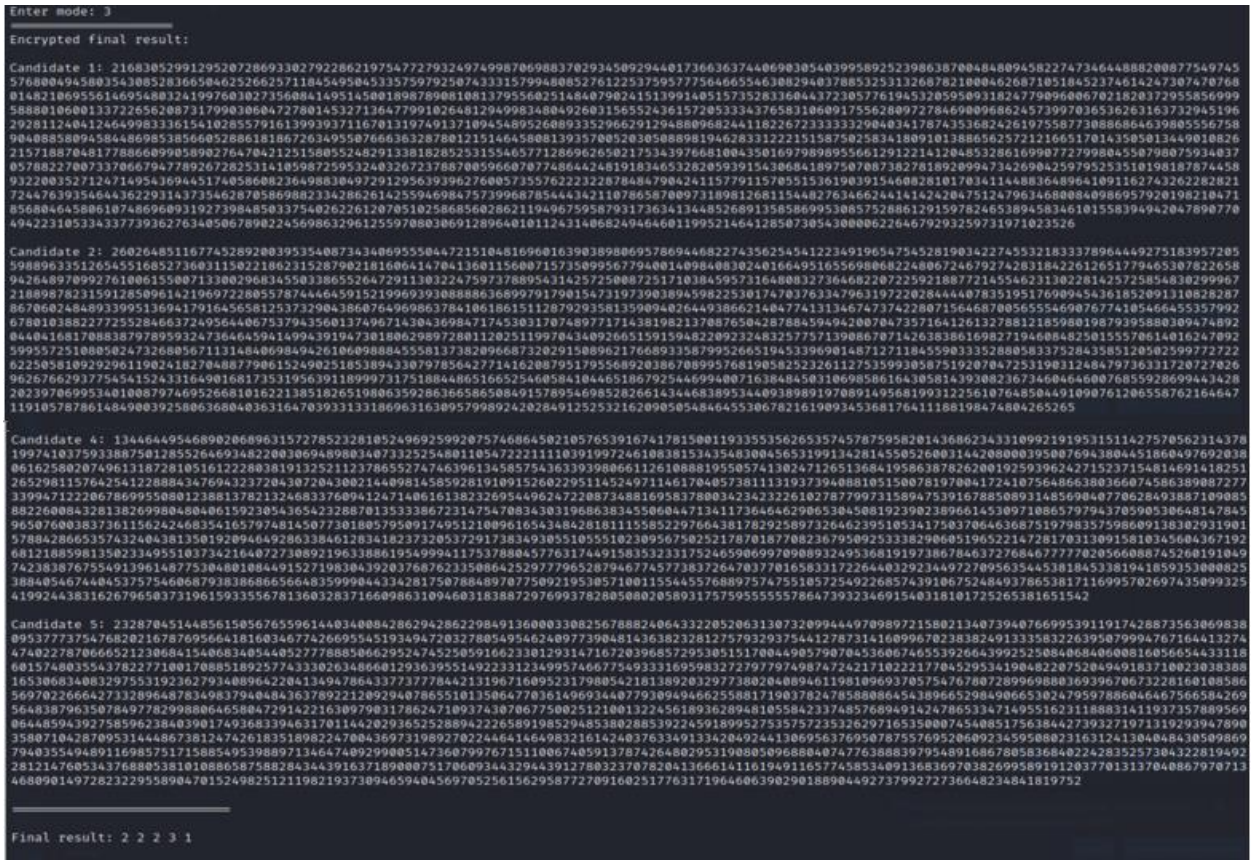
Hình 11: Kết thúc lượt vote



Hình 12: Kết quả mã hóa được lưu tại file 'resultElection.txt'

Mỗi dòng tương ứng với kết quả của từng ứng cử viên sau khi tổng hợp. Vì kích thước key là 3072 bits, nên ciphertext sau khi mã hóa tương đối lớn.

Tiếp theo là tới hàm **View Result**, đây chính là quá trình **Decryption**. Chương trình sẽ lấy kết quả được lưu từ file '**resultElection.txt**' và đem đi giải mã, sau đó in ra màn hình cả ciphertext và plaintext.

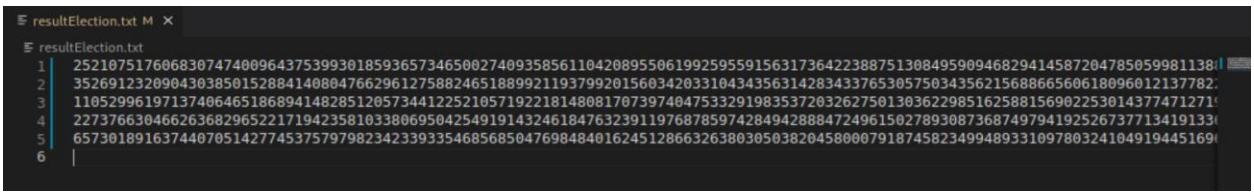


Hình 13: Xem kết quả bầu cử tại thời điểm hiện tại

Có thể thấy, sau khi trải qua lượt vote gồm 3 cử tri trên, kết quả bầu cử đã có là '2 - 2 - 2 - 3 - 1', tiếp tục thực hiện một lượt vote với 1 cử tri để có thể xác nhận tính chính xác của chương trình.

Hình 14: Bắt đầu lượt vote mới

Sau khi thực hiện mã hóa và tổng hợp, kết quả được lưu tại file '**resultElection.txt**' đã được cập nhật mới.



Hình 15: Kết quả được cập nhật vào file 'resultElection.txt'

Và thực hiện giải mã chúng ta nhận được một kết quả chính xác.

'2 - 2 - 2 - 3 - 1' + '1 - 0 - 0 - 0 - 1' = '3 - 2 - 2 - 3 - 2'

Hình 16: Xem kết quả bầu cử tại thời điểm hiện tại

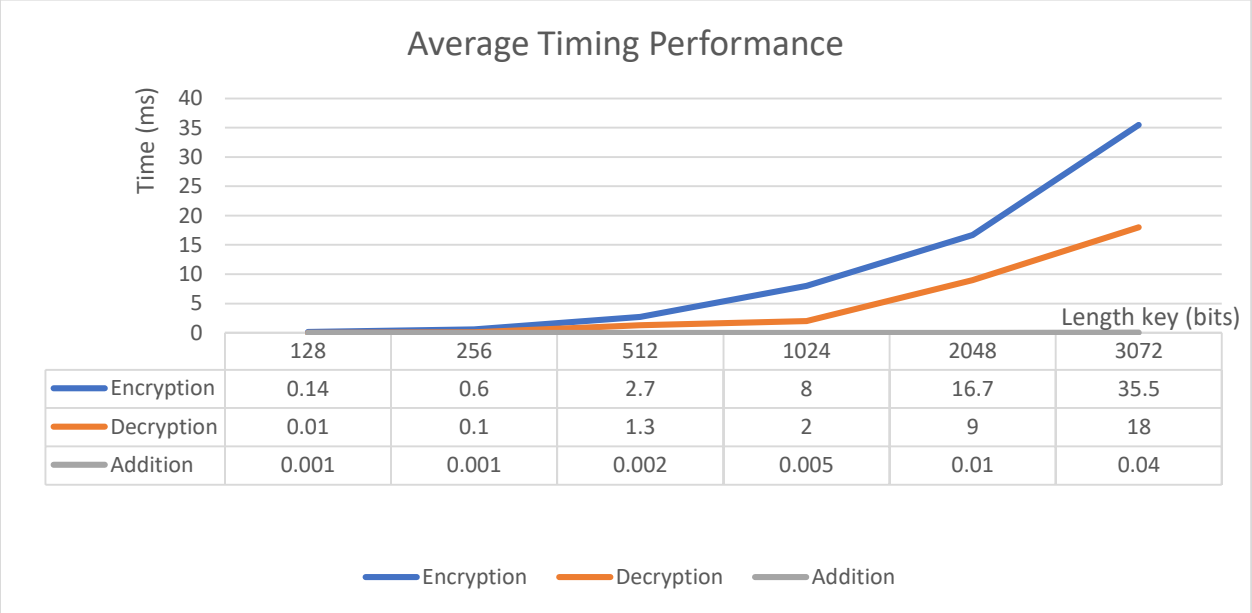
B. Timing Performance

Bên cạnh việc tính toán, nhóm em có thực hiện một chương trình giúp hỗ trợ việc tính thời gian của thuật toán và chia làm 3 phần: encryption, decryption, addition.

CPU:	Intel Core i5 Coffee Lake, 8265U, 1.60 GHz
RAM:	8 GB, DDR4 (On board +1 khe), 2400 MHz
Ổ cứng:	SSD 512 GB NVMe PCIe, Hỗ trợ khe cắm HDD SATA
Màn hình:	15.6", Full HD (1920 x 1080)
Card màn hình:	Card rời, NVIDIA GeForce MX250 2 GB

Hình 17: Cấu hình máy

Với cấu hình máy như trên, nhóm em đã thực hiện đo thời gian trên bộ key 128 – 256 – 512 – 1024 – 2048 – 3072 bits với 10000 vòng tương ứng với mỗi phần. Dưới đây là biểu đồ thời gian trung bình của từng phần (1 vòng):



	128	256	512	1024	2048	3072
Encryption	0.14	0.6	2.7	8	16.7	35.5
Decryption	0.01	0.1	1.3	2	9	18
Addition	0.001	0.001	0.002	0.005	0.01	0.04

TÀI LIỆU THAM KHẢO

[1] S. Saproo, V. Warke, S. Pote and R. Dhumal, "Online Voting System using Homomorphic Encryption," in *International Conference on Automation, Computing and Communication 2020 (ICACC-2020)*, 2020.

[2] S. M. Anggriane, S. M. Nasution and F. Azmi, "Advanced e-voting system using Paillier homomorphic encryption algorithm," *International Conference on Informatics and Computing (ICIC)*, pp. 338-342, 2016.

[3] B. Raut, M. Jagtap, S. Ghule, K. Jadhav and P. S. P. Aundhakar, "Homomorphic Encryption Based Online Voting System," *International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET)*, vol. 6, no. 6, pp. 159-163, 2019.

[4] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *IN ADVANCES IN CRYPTOLOGY — EUROCRYPT 1999*, Springer-Verlag, 1999, pp. 223-228.

[5] T. Sridokmai and S. Prakancharoen, "The homomorphic other property of Paillier cryptosystem," in *2015 International Conference on Science and Technology (TICST)*, 2015.

[6] M. H. A. Z. A. &. I. Z. Murtaza, "Blockchain Based Anonymous Voting System Using zkSNARKs," in *International Conference on Applied and Engineering Mathematics (ICAEM)*, 2019.