A Secure Cryptographic Election Model

Ochieng' Daniel Achola[1], Oleche Paul[2], Opiyo Richard[3] and Oluoch Nyamwala[4]

[1-3]Department of Pure & Applied Mathematics, Maseno University, Kenya
[4]School of Biological & Physical Sciences, Moi University, Kenya
[1]kazidane2003@yahoo.com, [2]poleche@gmail.com, [3]opiyorichard@yahoo.com, [4]foluoch2000@yahoo.com

Abstract– **Political reform is needed to reverse the dwindling voter apathy which is a growing concern in many democracies. Simplification of election procedure through introduction of e-voting is one measure that can be undertaken even though it has considerable potential for large scale fraud due to automation and network connectivity. Employment of e-voting scheme to conduct large scale multi-hierarchical election in a secure way is feasible provided certain deficiencies of existing voting protocols are addressed. In this paper, we propose a hybrid cryptographic voting protocol with a stronger audit trail. We have used cryptographic techniques including homomorphic secret sharing for universal verifiability to publish results in bulletin boards, zero-knowledge proofs in proving correctness of permutations in mixnets and validated votes using blind signatures to encrypt incoercible protocols. We have also curbed coercibility through receipt-freeness in deniable encryptions with randomness jointly chosen by the voter and tamper resistant tokens.**

***Keywords*– Voting Protocols, Zero Knowledge Proofs, Blind Signatures, Universal Verifiability and Homomorphic Secret Sharing**

I. BACKGROUND INFORMATION

Elections have long history, ancient Greece male landowners voted in "negative elections" that were recorded on broken pieces of porcelain, [15]. Venice on the other hand introduced approval voting with every contestant receiving a thumb up or down that saw the winner receives the highest votes. For increased error prevention, United States used multiple voting clerks who independently recorded voters' preferences by simple acclamation. It was, however, replaced by the introduction of ballot boxes in the 18th Century by individuals or parties and eventually party tickets. Aussies curbed external influence and audited secret ballot elections by using state printed ballot boxes. These were safely kept until Election Day for distribution to each eligible voter who then voted in isolated booths.

In New York booths were made with lever machines arranged in grids with rows indicating the seat contested while political parties were indicated by columns. Modern electronic voting can, however, be traced back to the introduction of Direct Recording by Electronics [DRE] with a special voting software which strictly denied access to personal computer based connectors like Universal Serial Bus (USB). They were perfect except for lacking tamper proof audit trail, thus the fear of production of erroneous results due to bugs and malicious codes that would go undetected [14]. Voter Verified Paper Audit Trail [VVPAT] introduced by Mercuri, [16] 1992 curbed erroneous production of results by printing out entire ballot on a scrolling receipt visible to the voter behind the glass which the voter confirms or rejects. There is strong empirical evidence that voting fraud is a regular occurrence through incentives and moreover, there isn't enough assurance to the voters and eventual prevention from nearly all future direct verification. In elections, any participant is potentially highly motivated to perform fraud. Election is particularly a very complicated process that requires public audit and significant amount of secrecy. These conflicting requirements including auditing partially secret process, failure detection and recovery, adversaries, incentives, verifiability and secrecy in advancing democracies call for cryptography as the alternative measure to offering provable security with stronger audit trail.

II. PRELIMINARIES

A) Cryptosystem

Definition 1: *A cryptosystem is a quintuple*

S = (P, C, K, E, D) *such that*

(i) P, C, K are sets with P as plaintext, C as cipher text and K as key space

(ii) $E = \{E_k \mid k \in K\}$ *is a family encryption functions and*

(iii) $D = \{D_k \mid k \in K\}$ *is a family of decryption function*

Cryptosystems have long fascinating history ranging from monoalphabetic schemes all the way to public key cryptosystems. Some of the classical schemes include Shift Cipher which are of the form

$$d_k(e_k(x)) = x: \quad \forall_x \in \square_{26} \tag{1}$$

For a particular key *k*=3 is often called the Ceaser cipher. Hill Cipher cryptosystem encrypted by *Kmxm* matrix as the key

$$C = (E_k(P)) = PK$$

and could be decrypted by use of inverse matrix

$$P = (D_k(C)) = CK^{-1}$$

Decryption is only possible if *K* has an inverse. The above ciphers were considered unbreakable until Claude Shannon introduced the concept of perfect secrecy in his paper "Communication theory of secrecy systems"

Theorem 2.2 Perfect Secrecy

Let S= {P, C, K, E, D} be a cryptosystem with $|C| = |K| = |P|$ *and* $P_p > 0$ *for each* $p \in P$ *then S guarantees perfect secrecy iff*

(i) for each $p \in P$ *and for each* $c \in C$ *there exists a unique* $k \in K$ with $\mathrm{E}_k(P) = c$

(ii) the keys in K are uniformly distributed and used with probability of $\frac{1}{|K|}$

Shannon further introduced the concept of entropy (mathematical measure of information) that is computed as a function of probability distribution

B) Entropy

Definition 3 *Let X be a random variable that takes a finite set of values* $X = x_i \dots x_n$ *for each* $1 \le i \le n$ *with the probability distribution* $P_i = P(x = x_i)$ *then the entropy of X is given by*

$$H(X) = -\sum_{i=1}^{n} p_i \log \mathrm{p}_i \tag{2}$$

and by Jensen's inequality it follows that

$$H(X) = -\sum_{i=1}^{n} p_i \log \mathrm{p}_i = \sum_{i=1}^{n} p_i \log \frac{1}{p_i} \le \log \sum_{i=1}^{n} (p_i \frac{1}{p_i}) = \log\ n \tag{4}$$

As Seneca, a Roman philosopher rightly puts it "Eternal law has arranged nothing better than giving one way into life but many ways out", [17]. The concept of public key cryptography is elegant but simple. It was first suggested by Diffie & Hellmann [20] who fixed a generator $g \in G$ in a finite cyclic group $G(G = (\square_p))$ or a point in elliptic curve.

C) Public Key Cryptosystem & Indistinguishable Security

Definition 4 *A public key cryptosystem is a set of Uniform Probabilistic Polynomial Time (PPT) algorithm G,E,D such that given the security parameter k the following operations are defined:*

(i) Key pair generation G implies using the public key algorithm G to generate a matching public key and secret key

$$(\xi, \gamma) \leftarrow G(1^k)$$

(ii) Encryption E implies using the public key encryption algorithm E to encrypt the plaintext p. This process is usually randomized using the randomization value $r \in R_\xi$.

$$c = E_\xi(p, r)$$

(iii) Decryption D implies using the secret key γ *and the decryption algorithm D*

$$p = D_\gamma(c)$$

We decrypt a cipher text c in the cipher text space c_ξ

Rivest, Shamir and Adleman in their *RSA* scheme based on prime factorization of integers were the first to implement a practical public key cryptosystem based on trapdoor function,[18]. In particular RSA uses the multiplicative group

$$\square_k = 1 \le i \le k-1 \quad and \quad \gcd(i,k) = 1$$

Theorem 2.5 *Let (n,e) be the public key* ξ *and* γ *be the private key, then for each plaintext* p *with* $0 \le p \le n$

$$p = (p^e)^d \bmod\ n$$

Definition 6 *A public key cryptosystem G,E,D is said to be indistinguishably secure under chosen plaintext attack(IND-CPA) if there exist a negligible function v (.) such that for all the Adversary in non-Probabilistic Polynomial Time* (PT^*)

$$\begin{aligned}&\Pr[(\xi,\gamma) \leftarrow G(1^k); (\mathrm{P}_0,\mathrm{P}_1,\mathrm{state}) \leftarrow \\ &\mathrm{Adversary(choose,\xi)b} \leftarrow \{0,1\}; \mathrm{c} \leftarrow \mathrm{E}_\xi(\mathrm{P}_\mathrm{b}); \\ &\mathrm{b'} \leftarrow \mathrm{Adv(guess,\ c,\ state)b{=}b'}] < \frac{1}{2} + v(k)\end{aligned}$$

Letting *g* the generator of q-order subgroup $p \in \square_p$ (a point in (($\square_2$ x $\square_2$ a point in elliptic curve) with a primitive element $y \in P$ and the possibility of an adversary obtaining a decryption of a few chosen cipher text before receiving the challenge cipher text, indistinguishable security under chosen cipher attack (IND-CCA) is proposed [18].

Definition 7 *A public key cryptosystem G, E, D is said to be indistinguishably secure under chosen cipher text attack 1 (IND-CCA1) if there exist a negligible function v(.) such that for all the* $Adv \in PT^*$ *given the decryption oracle ODec (.)*

$$\begin{aligned}&\Pr[(\xi,\gamma) \leftarrow \mathrm{G}(1^k); (\mathrm{P}_0,\mathrm{P}_1,\mathrm{state}) \leftarrow \\ &\mathrm{Adv}^{\mathrm{ODec}_\gamma}\mathrm{(choose,\xi)b} \leftarrow \{0,1\}; \mathrm{c} \leftarrow \mathrm{E}_\xi(\mathrm{P}_\mathrm{b}) \\ &\mathrm{b}' \leftarrow \mathrm{Adv(guess,c,state)b{=}b'}] < \frac{1}{2} + v(k)\end{aligned}$$

For all *p* in chosen cipher text attack, the public encryption scheme *G,E,D* is IND-CCA 1 is secure if it holds for every *PPT* adversary A

$$\mathrm{Adv}_{\text{IND-CCA1}}(\mathrm{A}) = |\Pr[\text{IND-CCA1=1}]-\frac{1}{2}|$$
$$\leq \mathrm{negl(k)}$$

The Adversary A has no access to the decryption oracle under the IND-CCA 1. We make it harder for the attacker to extract any information about the cipher text upon getting access to the decryption oracle *ODec* (.) that can be queried by using IND-CCA2 security.

Definition 8 *A public key cryptosystem G, E, D is said to be cipher text indistinguishably secure under chosen cipher text attack (IND-CCA2) if there exists a negligible function v(.) such that for all the* $Adv \in PT^{*}$ *given the decryption oracle ODec(.)*

$$\Pr[(\xi,\gamma) \leftarrow \mathrm{G}(1^{k});(\mathrm{P}_0,\mathrm{P}_1,\mathrm{state}) \leftarrow$$
$$\mathrm{Adv}^{\mathrm{ODec}_{\gamma}(.)}(\mathrm{choose},\xi)\mathrm{b} \leftarrow \{0,1\};\mathrm{c} \leftarrow \mathrm{E}_{\xi}(\mathrm{P}_{\mathrm{b}})$$
$$\mathrm{b}^{'} \leftarrow \mathrm{Adv}^{\mathrm{O\,Dec}_{\gamma}(.)}(\mathrm{guess,\ c})\mathrm{b} = \mathrm{b'}]<\frac{1}{2}+\mathrm{v(k)}$$

Without the secret key γ no one would be able to generate cipher text whose plaintext is related to that of *c*. The PKCS is IND-CCA2 secure provided the signature is existentially Unforgeable.

Definition 9 *A public key cryptosystem G,E,D is said to be cipher text indistinguishably secure under replayed/relaxed chosen plaintext attack (IND-RCCA) if there exists a negligible function v(.) such that for all the* $Adv \in PT^{*}$ *given the decryption oracle ODec(.)*

$$\Pr[(\xi,\gamma) \leftarrow \mathrm{G}(1^{k});\ (\mathrm{P}_0,\mathrm{P}_1) \leftarrow$$
$$\mathrm{Adv}^{\mathrm{ODec}_{\gamma}(.)}(\mathrm{choose},\ \xi)\mathrm{b} \leftarrow \{0,1\};\mathrm{c} \leftarrow \mathrm{E}_{\xi}(\mathrm{P}_{\mathrm{b}})$$
$$\mathrm{b}^{'} \leftarrow \mathrm{Adv}^{\mathrm{ODec}_{\gamma}\mathrm{P}_0\mathrm{P}_1(.)}(\mathrm{guess,\ c})\mathrm{b} = \mathrm{b'}]<\frac{1}{2}+\mathrm{v(k)}$$

Combining the two cipher texts and using the algebraic property of the homomorphic public key cryptosystem the resulting cipher text encodes a combination of underlying plaintexts under specific group operation for addition or multiplication. Indistinguishable security under replayed chosen plaintext attack (IND-RCCA) therefore provides a middle ground between IND-CPA and IND-CCA that specifically allows adversaries to generates fresh cipher text c' from existing cipher text *c* such that

$$\mathrm{D}_{\gamma}(\mathrm{c})=\mathrm{D}_{\gamma}(\mathrm{c'})$$

Definition 10 *A public key cryptosystem G, E, D is said to be homomorphic for binary relations* $(\oplus,\otimes)$ *if*

(i)For all $(\xi,\gamma) \leftarrow G(1^{k})$ *given the message* $P_{\xi};(P_{\xi},\oplus)$ *forms a group*

(ii) For all $(\xi,\gamma) \leftarrow G(1^{k})$ *given the cipher text* $C_{\xi};(C_{\xi},\otimes)$ *forms a group*

(iii)For all $(\xi,\gamma) \leftarrow G(1^{k}),\ \forall(\mathrm{c}_1,\mathrm{c}_2) \in \mathrm{C}^{2}_{\mathrm{PKCS},\xi}$

Defining $\otimes$ as the element wise product of a cipher text pairs.

$$D_{\gamma}(c_1 \otimes c_2) = D_{\gamma}(c_1) \oplus D_{\gamma}(c_2)$$

And using the homomorphic property of a cryptosystem, we create a different cipher text $c^{'}$ that encodes the same text as *c* Public Key Cryptosystem is homomorphic for $(\oplus,\otimes)$ if $(P_{\xi},\oplus)$ forms a group, then there exists identity plaintext P_0 such that $P_{\xi},\ \mathrm{p}+\mathrm{p}_0=\mathrm{m}$.

And defining re-encryption algorithm as

$$\mathrm{RE}_{\xi}(\mathrm{c,r})=\mathrm{c} \otimes \mathrm{E}_{\xi}(\mathrm{p}_0,\mathrm{r}) \qquad (5)$$

if

$$\mathrm{D}_{\gamma}(\mathrm{c}) = \mathrm{m} \quad \text{then} \ \mathrm{D}_{\gamma}(\mathrm{RE}_{\xi}(\mathrm{c})) = \mathrm{m} \qquad (6)$$

Because of the malleability of the cipher text in homomorphic cryptosystems, security is limited to re-encryption. El Gamal scheme encrypted as

$$\mathrm{c} = (\mathrm{g}^{\mathrm{r}},\ \mathrm{p},\ \mathrm{y}^{\mathrm{r}})$$

El Gamal is homomorphic for $(\oplus,\otimes)$

$$(\mathrm{g}^{\mathrm{r}_1},\mathrm{p}_1,\mathrm{y}^{\mathrm{r}_1}) \otimes (\mathrm{g}^{\mathrm{r}_2},\mathrm{p}_2,\mathrm{y}^{\mathrm{r}_2})=(\mathrm{g}^{\mathrm{r}_1+\mathrm{r}_2},(\mathrm{p}_1+\mathrm{p}_2),\mathrm{y}^{\mathrm{r}_1+\mathrm{r}_2}) \qquad (7)$$

Equation (7) exhibits homomorphism and is IND-CPA 2 secure and Exponential Elgamal is given as

(i) Key Generation

We select a prime *p* such that another large prime *q* divides $(p-1)$ we further select another generator *g* of q - order subgroup of $\square_p$.

$\mathrm{P}_{\xi}=\square_{\mathrm{p}} \quad \gamma = \mathrm{x}$ Where *x* is randomly selected in $\square_p$

$\xi = \mathrm{y} = \mathrm{g}^{\mathrm{x}} \mod \mathrm{p}$

(ii) Encryption is given by

$$\mathrm{E}_{\xi}(\mathrm{p,r})=(\alpha_2,\alpha_2)=(\mathrm{g}^{\mathrm{r}},\mathrm{g}^{\mathrm{p}},\mathrm{y}^{\mathrm{r}}) \bmod \ \mathrm{p}$$

(iii) Decryption is obtained by discrete algorithm

$$\mathrm{D}_{\gamma}(\alpha_1,\alpha_2) = \log\,[\frac{\mathrm{m}\beta^{a}}{\mathrm{y}^{\mathrm{a}}}] \bmod \mathrm{p}$$

on performing element wise multiplication on cipher text pairs of Elgamal, Letting $\mathrm{H}_1=\mathrm{E}_{\xi}(\mathrm{p}_1,\mathrm{r}_1)$ and $\mathrm{H}_2=\mathrm{E}_{\xi}(\mathrm{p}_2,\mathrm{r}_2)$ we obtain

$$\begin{aligned} E_{\xi}(p_1,r_1)\otimes E_{\xi}(p_2,r_2) &= (g^{r_1},g^{p_1},y^{r_1})\otimes(g^{r_2},g^{p_2},y^{r_2}) \\ &= (g^{r_1+r_2},g^{p_1+p_2},y^{r_1+r_2}) \\ &= E_{\xi}(p_1+p_2,r_1+r_2) \end{aligned}$$

And performing encryption in Paillier scheme, as

$$c = g^{p}r^{n} \bmod n^{2} \qquad (8)$$

Indeed, Paillier is homomorphic for $(+,\times)$ over the plaintext space $□_n$

$$\begin{aligned} E_{\xi}(p_1,r_1)\times E_{\xi}(p_2,r_2) &= (g^{p_1},r_1^{\,n})\times(g^{p_2},y^{r^{n}_{2}}) \\ &= (g^{p_1+p_2},(r_1r_2)^{n}) \\ &= E_{\xi}(p_1+p_2,\ r_1r_2) \end{aligned}$$

and for a generalized Paillier, we have

$$E_{n,s}^{\ \ Pai}(p) = g^{p}r^{n^{s}} \bmod n^{s+1} \qquad (9)$$

Definition 11 Secret Sharing

A perfect secret sharing scheme realizing the access structure Γ *is a method of sharing a key K amongst a set of* Ω *participants in such a way that*

(i)If authorized subsets of participants $B\subseteq\Omega$ *pool their shares, they can determine the value of K*

(ii) If unauthorized subsets of participants $B\subseteq\Omega$ *pool their shares they cannot determine the value of K*

To make the secret key γ secure in electronic voting, the trustees generate key pair together with no single party learning the complete secret key in the process. The secret key γ is availed to a quorum of trustees *t* with each trustee *i* having a share of the secret key γ^{i} Shamir [13] in his lemma introduced the concept of secret sharing protocol that is crucial to the implementation of threshold cryptography.

Lemma 12 *Let* $l,t\in□$, $l\le t$ *. Also let where* $x_i,y_i\in□/p□$, $1\le i\le l$ *be pairwise distinct. Then there are exactly* p^{t-1} *polynomials* $b\in(□/p□)[X]$ *of degree* $<t-1$ *with* $b(x_i)=y_i$ $1\le i\le l$, *[13]*

Benaloh showed how to achieve verifiable secret sharing if one-way function exist which tolerates minority of colluders using error-correction codes. Yvo and Yair [21] proposed El Gamal for efficient threshold scheme given its algebraic structure which is such that each secret key x_i is associated with the public key $y_i = g^{x_i}$ and obtain the decryption

$$P=\left(\frac{\beta}{\prod\alpha^{x_i\lambda_i(0)}}\right) \qquad (10)$$

where $\alpha^{x_i}=\gamma$ is computed by each trustee independently. We therefore achieve a distributed generated key with no single party, electoral board or even the dealer learning the complete secret key.

Definition 13 Digital Signature

A digital signature scheme is a tuple (Gen, Sign, and Ver) where the following conditions are satisfied

- $Gen(1^{k})$ *is PPT algorithm that takes security parameter K and outputs verification/signature keys* (Ver_k,Sg_k),
- $Sign_{Sgk}(m)$ *is a PPT algorithm that takes signature scheme* (Sg_k) *a message m and outputs a signature* σ
- $Ver_{vk}(m,\sigma)$ *is a PPT algorithm taking as input verification key* (vk) *a message m and a signature* σ *and outputs a bit* $b\in\{1,0\}$

We say that the signature scheme (Gen, Sgn, Ver) is correct iff it holds for all messages m that

$$\Pr[Ver_{vk}(m),Sg_{Sgk}(m) = 1:(Ver_{vk},Sg_{Sgk})=G(1^{k})] > 1\text{-negl}(k)$$

Moreover we require that Existential Unforgeability under one time chosen Plaintext Attack (EUF-CPA)

D) Perfect Zero Knowledge Proof

Goldwasser & Micali, [18] introduced the concept of zero knowledge proof an interaction algorithm between a Prover *P* and a verifier *V* resulting into the validity of the assertion for verifiable voting protocol without learning anything other than the truth of the assertion.

Definition 14 *An interactive protocol <P, V> for a language L is defined as a perfect zero knowledge proof (ZKP) if there exist a negligible function v(.) such that the protocol has the following properties*

(i)Completeness

$$\forall_{x}\in L,\Pr[Out_{v}<P_{(x,y)},V_{(x)}>=1]>1\text{-}v(k)$$

(ii)Soundness

$$\forall P^{*},\forall_{x}\in L\ \ \Pr[Output_{v}<P_{x}^{\ *},V_{x}>=1]<\frac{1}{2}$$

(iii)Zero Knowledge

$$\exists PPT\ \ S,\ \forall V^{*},\forall_{x}\in L$$

$$S(x)\ □\ Output_{V^{*}}<P_{(x,w)},V^{*}(x)>$$

Where $L,x,w,S,PPT,P,P^{*},V,V^{*}$ symbolizes Language, Strings, Witness, Simulator, Uniform Probabilistic Polynomial Time, Prover, Dishonest Prover, Verifier and Dishonest Verifier respectively. Verifiable voting protocol involves setting up election parameters, preparing ballot with special machines that encrypts votes, posting encrypted ballot in a readable bulletin board for ballot recording, running an algorithm and consequently posting results on bulletin board and production of results for public verification. In casting of the votes by voter N_j in an election ε with R_k races having O_k options to choose from, we need l_i officials to serve N voters. In general for voter N_j selection for races R_k denoted by $(t_j^{\,k})$ we obtain the N_j ballot

$$P_j = (t_j^{\,1}, t_j^{\,2} \ldots t_j^{\,s}) \qquad (11)$$

ε defines election parameters including ξ and γ shares $\gamma^1 \cdots \gamma^l$ where $\gamma^i \in l_i$. N_j casts a plaintext ballot P_j encrypted in c_j using a randomized value r_j under ξ with the encryption algorithm depending on the encoding mechanism of equation (10) above. N_j should prove that her vote correctly encodes her intent without coercion from an adversary. A secret receipt for personal verifications therefore issued.

Okamoto, [17] developed receipt freeness for blind signature. Benaloh, [3] developed the first simple practical additive homomorphic scheme with a "yes" vote encrypted as '1' and a "no" vote '0' and anyone could use ξ to tally but only the election officials l_i is allowed to perform threshold decryption. In modern day democracy with multiple candidates involved in multiple races we can efficiently use Pallier Cryptosystems in equation (8) decryption. Chaum [6] in his Mixnet presented protocols that generate shuffled lists of messages sent by set of senders each with ξ but keeping the γ secret.

$$c_{0,j} = E_{\xi_1}(r_{1,j}, E_{\xi_2}(r_2, j)\ldots E_{\xi m}(r_m, j)) \qquad (12)$$

A re-encryption Mixnet that combines algebraically re-randomized Mixnet and generates γ

$$\begin{aligned} \gamma_i &= x_i \leftarrow Z_{p-1} \quad :\text{p is a prime} \\ \xi_i &= y_i = g^{x_i} \bmod p \end{aligned} \qquad (13)$$

Recalling Elgamal cipher text

$$c = E_\xi(p,r) = (\alpha_1,\alpha_2) = (g^r,p,y^r)$$

Considering the Mixnet joint public key

$$Y = \prod_{i=1}^{1} \xi_i = g\sum_{i=1}^{1} x^i \qquad (14)$$

and with the ability to re-encrypt with Elgamal

$$RE_\xi(c,r') = (\alpha g^{r'}.\beta y^{r'}) = E_\xi(c, r+r')$$

gives the encryption of the plaintext under the joint public key

$$c_{0,j} = E_\xi(p_j,r_j) \qquad (15)$$

Re-encrypting equation (14) above with fresh randomness, we obtain re-encrypted cipher text

$$c_{i,j} = RE_\xi(c_{(i-1,j)},r_{(i,j)}) \qquad (16)$$

we make it secure by

$$c^* = (\alpha,g^{r^*}, p^*, \beta.y^{r^*})$$

and check $p_0 = p^*.p_1$ for dishonest mix-servers.

E) Elliptic Curves & Mixnet

Definition 15 *Let* $p > 3$ *be a prime. The elliptic curve* $y^2 + x^3 + ax = b$ *over* $\square_p$ *is a set of solutions* $x, y \in \square_p$ x $\square_p$ *to the congruence*

$$y^2 \equiv x^3{+}ax{+}b \mod p$$

where $a, b \in \square_p$ *are constants such that* $4a^3{+}27b^2 \neq 0$ *together with a special point O called point of infinity.*

Theorem 2.16 *Let E be an elliptic curve defined over* $\square_p$ *where p is a prime, p>3 then,* $\exists n_1, n_2 \in \square$ *such that E is isomorphic to* $\square n_1$ x $\square n_2$. *Further* n_1, n_2 *and* $n_2 \,|\, (p-1)$. *If* n_1,n_2 *can be computed then, E is a cyclic group isomorphic to* $\square_n$ *that can be used to set up Elgamal cryptosystem.*

Elliptic curve groups for cryptography are examined with the underlying prime number fields F_p with p > 3 and $F_2^{\,m}$ (a binary representations with 2^m elements. David Chaum, [7] recently introduced mixnets that shuffles and decrypts encrypted voters. Mixnets are flexible, require small computational ability, and can be verified by public since the authorities can easily prove the correctness of their procedures. A model Mixnet has

(i) the election policy committee that determines the parameters (q, E ,g) that are used in El Gamal cryptosystem on elliptic curve E , q being the prime order of elliptic curve E and g as the generator.

(ii) The shuffling management announces authorized parameters (q,E,g) to the shuffling centers. The j^{th} shuffling center SC_j randomly chooses $\gamma = x_j \bmod q$ and $\xi = y_i = [x_j]g$ to generate the proof

$$\xi' = y'_i = [\beta_j]g \qquad ; \ \beta_j \in \square/q\square$$

(iii) The shuffling management verifies each

$$\xi = y'[\,j = 1\ldots m]$$

as

$$c_j = \mathrm{H}(p,q,g,y_j y_j')$$

$$\xi' = y_j = [r_j]g\text{-}[c_j]y_j, \quad ;y_j \in E,\ y_j \neq 0$$

the verified public keys ξ' are combined to form a common public key

$$\Xi = \sum_{j=1}^{m} \xi$$

The voter N_j will use the parameter Ξ and (q, E, g) certified by the election committee to encrypt his vote (m_i)

$$\mathrm{G_i M_i} = ([r_j]g,\ m_i + [r_i E]) \qquad (17)$$

Where r_i the element is randomly generated by the voter N_j and ID_j is the information that identifies the voter who may prove the knowledge of his vote m_i. The voting center sends the list of accepted votes from each voter $(\mathrm{G_i M_i})_{i=1\ldots n}$ to shuffling management center that verifies each component in E by computing

$$\varsigma = \sum_{j=1}^{n} [c_j]\ \mathrm{G}_j'$$

III. OBJECTIVES

In this paper we model a secure and verifiable hybrid cryptographic election that offer

(i) Provable secure election scheme that can be trusted in electoral process

(ii) Verifiable secret ballot elections scheme that can be publicly audited and independently verified for voter assurance

IV. MODEL

A) Threshold Secret Sharing

Let γ be the secret key divided into n pieces $\gamma_1, \gamma_2, \ldots \gamma_n$ and distributed amongst n shareholders in such a way that for any threshold value t the knowledge of any t or more γ_i makes γ easily computable but $t-1$ or fewer γ_i leaves γ completely undetermined.

B) Probabilistic Encryption

We prevent information leakage by developing an encryption algorithm E_k. We chose large and distinct primes p and q with $N = pq$ and further select a larger prime r greater than the size of secret domain. We find some non-quadratic residue γ such that the Legendre symbol satisfy $\frac{\gamma}{p} = \frac{\gamma}{q} = 1$ Hence Jacobi Symbol $\frac{\gamma}{N} = 1$ Then, the public key pair is $\xi(N, \gamma)$ while secret key pair is $\gamma(p, q)$. To use E_k to encrypt a share γ the dealer randomly selects $x \in \square_n$ and outputs a value $x^r y^s \bmod N$ the trapdoor factors (p, q) of N are needed to recover the original value γ from its encrypted value.

C) Homomorphism & Homomorphic Encryption

Letting (t, n) be secret sharing scheme, F_z a function that reconstructs γ from any subset of t or more shares with $\square \subseteq \{1, 2, \ldots n\}$ and $|z| \geq t$ with t a pre-specified threshold value

$$\gamma = F_z(\gamma_1, \gamma_2 \ldots \gamma_t) \qquad (18)$$

Taking the two binary operations $\oplus$ and $\otimes$ defined on elements of a secret domain and share domain. If (t, n) secret sharing has the property $(\oplus, \otimes)$ then,

$$\gamma = F_z(\gamma_1, \gamma_2 \ldots \gamma_t) \qquad \gamma' = F_z(\gamma'_1, \gamma'_2 \ldots \gamma'_t)$$

we can easily compute

$$\gamma \oplus \gamma' = F_z(\gamma_1 \otimes \gamma'_1, \gamma_2 \otimes \gamma'_2, \ldots \gamma_t \otimes \gamma'_t) \qquad (19)$$

Unlike Shamir polynomial based (t,n) secret sharing where the sum of shares of the secret are the shares of sum of the secret, homomorphic encryption has the ability of computing cipher text without decryption. The encryption function E_k is homomorphic if given $E(x)$ and $E(y)$ we obtain $E(x \oplus y)$ without decrypting x, y. We broadcast encrypted values to each shareholder to independently construct encrypted polynomial to compute encrypted shares for different shareholders and can be compared with encrypted value from the polynomial. We verify that the shares are collectively t-consistent in such a way that every subset of the threshold number of shares can construct the same secret.

D) Session Keys

We use forward secrecy which can only be accomplished via Diffie-Hellman protocol to prevent compromisation of past sessions for an adversary who gets the long lived keys. Suppose G is non-abelian group and $S, T \subset G, \quad [S, T] = 1$ Taking $s \in S$ and $b \in G$ and publishing b and $c = b^s$ while keeping s as secret key. Here $b^s = s^{-1}bs$ If we wish to send $p \in G$ as session keys, we first choose a random $t \in T$ and send $E = p^{(c^t)}$ with a header $h = b^t$ which the receiver calculates as

$$(b^t)^s = (b^s)^t = c^t$$

with the header. To compute $E^{'} = (c^t)^{-1}$ this allows one to decrypt the session keys

$$(p^{(c^t)})^{E'} = (p^{(c^t)})^{(c^t)-1} = p \qquad (20)$$

And we now use $x \in G$ as session keys.

E) IND-CCA2

We efficiently construct IND-CCA2 a one way semantically secure trapdoor function based on number assumptions. Using Dolev, Dwork & Naor, [10]. We transform Discrete Logarithm into IND-CCA2 secure scheme using one time signature. We consider a PPT adversary A having a negligible probability guessing the secret tag τ correctly if granted polynomial number of trials, yields statistical indistinguishability of either world 0 or world 1 (from a distribution with minimum entropy) further allowing adversary A to make decryption queries after getting challenging cipher text c'. We let the sign $\sigma = (\mathrm{Gen,Sign,Ver})$ be Existentially Unforgeable under Chosen Plaintext Attack (EUF-CPA) secure one time signature. We assume further that the verification key vk of Sign are elements of *q*-array alphabet $\sum^{\mathrm{n}}$(n large enough) then E_k computes a pair of $(\mathrm{vk,Sgk}) = \mathrm{Sign.G}(1^{\mathrm{k}})$ and sets τ (decryption queries) equated to instead of choosing vk uniformly at random. E_k computes $\sigma = \mathrm{Sig.sign}_{\mathrm{sgk}}(\mathrm{c}')$ and outputs a cipher text $\mathrm{c} = (\mathrm{c}',\sigma)$. We decrypt by checking if σ is a valid σ on c' using $\mathrm{vk} = \tau \in \mathrm{c}$

Construction

Let $E_\xi'(p,vk)$ given as $PKCS_2$ be IND-CCA1 secure provided $PKCS_1$ is IND - CPA secure. Then our scheme

$$\mathrm{PKCS_3{=}(Gen,Sign,Ver)}$$

must satisfy these conditions

(i) $\mathrm{Gen}\ (1^{\mathrm{k}})$: Compute $(\xi,\gamma) = \mathrm{PKCS_2.Gen}\ (1^{\mathrm{k}})$ and outputs (ξ,γ)

$$\mathrm{E}_\xi(\mathrm{p}) = \mathrm{Gen\ (vk,sgk)} = \mathrm{SIG.Gen}(1^{\mathrm{k}})$$

(ii) Encryption $\mathrm{c}' = \mathrm{E}'_\xi(p,vk)$

sign $\sigma = \mathrm{SIG.Sign_{sgk}c'}$ and outputs $\mathrm{c} = (\mathrm{c},\sigma)$

(iii) $\mathrm{D}_\gamma(\mathrm{c})$: Let $\mathrm{c} = (\mathrm{c}',\sigma)$ and $\mathrm{c}' = (\tau,\mathrm{c}_1,\mathrm{c}_2,\mathrm{c}_3)$ Set vk=τ

F) Deniable Encryptions

Assuming that the interested party (contestant) has the power to approach the voter or the electoral body, we propose additional security property. The Electoral body store data in a deniable way for multiparty computations using public key deniable encryption to curb coercion. If we let $\mathrm{E_k}$ be the encryption algorithm with public key ξ and a further public key faking algorithm ϕ given a bit b and random input r the resulting cipher text

$$\mathrm{c = E_k(b,r)} \qquad (21)$$

and the faking algorithm generates a fake random input $\rho = \phi\ \mathrm{(b,r,c)}$ Given b,ρ,c an adversary should be unable to distinguish

(i) ρ is uniformly chosen and $\mathrm{c = E_k(b,\rho)}$

(ii) c is generated as $\mathrm{E_k(b,r)}$ where r is independently and uniformly chosen and $\rho = \phi(b,\mathrm{r,c})$

Definition 1 *Let* $\exists$ *a family of* $\{\mathrm{S_t}\}_{\in\square}$ *where* $\mathrm{S_t} \in \{0,1\}^{\mathrm{t}}$ *together with the secret trapdoor information* d_t *such that:*

(i) $\mathrm{S_t}$ *is small:* $|\mathrm{S_t}| \le 2^{t-k}$ *for some sufficiently large* k

(ii) It is easy to generate a random element $c \in S_t$ *even without a secret* d_t

(iii) Given $c \in \{0,1\}^t$ *and* d_t *it is easy to decide whether* $c \in d_t$

(iv) Without d_t *values chosen uniformly from* S_t *are indistinguishable from values chosen* $\{0,1\}$

we can conveniently construct the translucent set $\{S_t\}$ *by letting* $t = s + k$ *and representing* $c \in \{0,1\}^t$ *as* $c = c_0, b_1, \ldots b_t$ *where* $c_0 \in \{0,1\}^s$ *and for* $i \ge 1$ *each* $b_i \in \{0,1\}$ *then,*

$$S_t = \{c_0, b_1, \ldots b_k \in \{0,1\} \mid \forall_i 1, \ldots k\} \qquad (22)$$

$$B(f^{-i}(c_0)) = b_i \qquad (23)$$

Here $|S_t| = 2^s = 2^{t-k}$ This construction is efficient given the trapdoor permutation on $\{0,1\}^s$ The length of the cipher text c is $t = s + k$ instead of $t = sk$ for public key encryption, we generate uniformly at random an element of translucent set $S_t \subset \{0,1\}^t$ with d as the corresponding private decryption key.

Encryption To encrypt 1 we send a random element of S_t, to encrypt 0 we send a random element of $\{0,1\}^t$

Decryption If $c \in S_t$ then output 1 else output 0.

Remark 2 For honest encryption, reveal true random choices used otherwise if the encrypted bit is 1 cipher text c is random in $\mathrm{S_t}$ and claim c was chosen at random $\{0,1\}^{\mathrm{t}}$ hence c is an encryption of 0. If the encrypted bit is 0, then deniability is possible because $c \in S_t$ with a negligible probability. The faking algorithm ϕ is possible in one direction but can be possible in many directions using parity scheme.

V. CONCLUSION

In this paper, we have constructed a secure hybrid cryptographic voting protocol considering different types of adversaries. We have used threshold, probabilistic and homomorphic encryption to construct secret keys. We have also constructed session keys to accomplished forward secrecy for adversaries who get long lived keys in advance. Considering probabilistic polynomial time adversary with negligible probability of guessing secret key, we have transformed Discrete Log scheme into IND-CCA2 secure one time signature based on number theoretic assumptions. And finally we have constructed deniable encryptions using public key faking algorithm to curb coercion from a powerful adversary who is able to approach election officials. We recommend further research on construction of secure tokens with zero knowledge proofs based on unique biometric properties.

REFERENCES

[1] Bandron O. Fouque P. Pointchavel D., Poupard G & Stern J.: Practical Multicandidate Election System. ACM Symposium on Principles of Distributed Computing. ACM pp. 274-283 (2001).

[2] Benaloh. J: *Verifiable Secret Ballot Elections* PhD Thesis. Yale University 1987

[3] Benaloh J: & Dwight Tuinsutra: *Receipt Free Secret Ballot Elections* In STOIC (4) page 544 - 553

[4] Canetti R., Dwork C., Naor M., & Ostrovosky R: Deniable Encryption. CRYPTO '97 LNCS 1294 Springer-Varlag pp. 90-104, 1997

[5] Cramer R., Gennaro R., & Shoenmakers B.: A secure & Optimally Efficient Multi-Authority Election Scheme EUROCRYPT '97, Springer-Verlag pp103-118 (1997)

[6] David Chaum: Secret Ballot Receipts, True Voter Verifiable Elections, IEEE Security & Policy (02):38-47(2004)

[7] David Chaum ...et al.: *Towards a Trustworthy Elections, New Directions in Electronic Voting*, Springer-Varlag, Berlin Heidelberg. (2010)

[8] David Chaum.: Blind Signatures for Untraceable Payments. CRYPTO '82 Plenum Press pp 199-203 (1982)

[9] David Kahn: *The Code breakers*, Macmillan, New York, USA (1973)

[10] Dolev, D. Dwork, C. & Naor M: Non Malleable Cryptography. SIAM Journal of Computer 30(2), pg 391-437 (2000).

[11] Fiat, A & Shamir, A.: How to Prove Yourself (Practical Solutions to Identifications & Signature Problems) CRYPTO '86 LNCS 263 Springer-Varlag pp 186-194 (1987).

[12] Fujioka A., Okamoto T. & Ohta K.: A Practical Voting Scheme For Large Scale Elections. AUSCRYPT '92 LNCS 718 Springer-Varlag pp. 224-251 (1993).

[13] Johannes A. Buchamann: *Introduction to Cryptography* Springer-Verlag, USA (2004).

[14] Lynn Landes: The Nightmare Scenario Is Here: Computer Voting With No Paper Trail, (August 2012), http://www.ecotalk.org/Dr Rebecca MercuriComputerVoting.htm

[15] Michael Shamos: Paper Vs Electronic Voting Records, http://www.electiontech.org, August (2012).

[16] Rebecca Mercuri: Voting Machine Risks, Commun. ACM 35(11):138 (1992).

[17] Richard A. Mollin: *Introduction to Cryptography, Second Edition*, Taylor and Francis, New York, USA (2007).

[18] Shafi Goldwasser & Mihir Belare: *Lecture Notes on Cryptography*, Aug 2001.

[19] Shannon E. Claude: A Mathematical Theory of Telecommunications. The Bell Systems of Technical Journal. Vol. 27, pp. 379-423, 623-656 July, Oct 1948.

[20] Whitefield Diffie & Martin E. Hellman: New Directions In Cryptography IEEE. Trans, Inform. Theory, IT-22:644-654, November 1976.

[21] Yvo Desmedt & Yair Frankel: Threshold Cryptosystems Lecture Notes in Computer Science. Springer Varlag, (1990).

APPENDIX

RSA Algorithm

(i) Choose some two prime numbers p and q

(ii) Calculate $n = \phi = pq$

(iii) Calculate $\varphi = (p-1)(q-1)$

(iv) $\prod(\mathrm{D_k},\mathrm{E_k}) = 1 \bmod \phi$

To calculate D_k we must choose E_k and the criteria for choosing E_k is given as

- E_k should be in the interval $1 < E_k < \phi$.
- E_k should be a prime number.
- E_k, ϕ be relatively prime, i.e. gcd (E_k, ϕ) = 1

(v) Calculate the value of D_k using extended Euclidian Algorithm table method. Euclidean Algorithm is

$$ax + by = \gcd\ (a,b)$$

In our case we let $a = \phi$ and $b = E_k$ to obtain

$$\phi(x) + E_k(y) = \gcd(\phi, E_k)$$

We obtain the value of D_k from the above equation as y

The two possible values of D_k are

- D_k should always be greater than ϕ

$$D_k > \phi \text{ then } D_k = D_k \bmod \phi$$

- D_k must always be positive

$$D_k < \phi \text{ then } D_k = D_k + \phi$$

(vi) Encryption of the plaintext (P) is given by

$$C = P^{E_k} \bmod \phi$$

.

(vii) Decryption of the cipher text (C) is given by

$$P = C^{D_k} \bmod \phi$$

El Gamal Algorithm

We generate El Gamal keys by randomly selecting *b* as the private key such that $\gamma = y^b \bmod p$ and sends it to officials who randomly chooses α_1, α_2 and encrypts the message

$$\alpha_1 = y^a \bmod p, \qquad \alpha_2 = m\gamma^a \bmod p$$

We obtain our encryption by

$$c = (\alpha_1, \alpha_2) = (g^r, p, y^r) \qquad (1.1)$$

Similarly our decryption is given by

$$p = \alpha_2 (\alpha_1)^{-b} \bmod p \qquad (1.2)$$

Example 1 *Let the dealer and officials agree on a prime number p=101 and a primitive element y=8 of 101. The officials chooses b=12 and computes a private key* $\gamma = 8^{12} \bmod 101 = 78$. *The dealer chooses a random exponent a=33 and computes* $\gamma^a = 78^{33} \bmod 101 = 92$ *suppose the dealer wishes to send the message m=53. To encrypt the message m they compute* $\alpha_1 = \gamma^a \bmod p = 8^{33} \bmod 101 = 51$ $\alpha_2 = m\gamma^a \bmod p = 53(92) \bmod 101 = 28$ *and sends the ciphertext (c=51,28) The officials decrypts it by computing*

$$\alpha_2 (\alpha_1)^{-b} = 28(51)^{-12} 53 \bmod 101$$

Secret Sharing

Lagrange's interpolation polynomial

$$b[x] = \sum_{i=1}^{k} y_i \Big(\prod_{j=1,\, j \neq i}^{k} \frac{x_j - x}{x_j - x_i} \Big) \qquad (1.3)$$

shows the existence of such polynomial that satisfies the value of *l* in the above lemma and corresponds to the point *i* as

$$b(x_i) = y_i \qquad 1 \le i \le l$$

and we obtain the linear system with the coefficient matrix a Vandermonde matrix whose determinant is given by

$$\det(U) = \Pi(x_i - x_j); \qquad i \le j \le l$$

where x_i and x_j are distinct by assumptions implying that the Kernel of the coefficient matrix has a rank $t-1$ resulting into p^{t-1} solutions to the linear system. Considering a special case where *t* = 1 we obtain one and only one such polynomial and we use it to solve the secret sharing problem. We chooses a prime number $p,\ p \ge n+1$ and a non-zero element $x_i \in$ □ / p□ to share a secret $s \in \{0, \ldots p-1\}$ by constructing the polynomial

$$b[x] = s + \sum_{j=1}^{t-1} b_j x_j \quad \text{of degree} \quad \le t-1$$

where $s = b_0$ is the secret and is obtained by

Example 2 *Let l=2*

$$b[x] = \sum_{i=1}^{2} y_i \Big(\prod_{j=1,\, j \neq i}^{2} \frac{x_j - x}{x_j - x_i} \Big)$$

$$= y_1 \frac{x_2 - x}{x_2 - x_1} + y_2 \frac{x_1 - x}{x_1 - x_2} \qquad (1.4)$$

Replacing *x* with x_1 and x_2 in the equation (1.4) above, we obtain

$$b[X]|_{x=x_1} = y_1, \qquad b[X]|_{x=x_2} = y_2$$

Without loss of generality

$$b[x]_{x=x_i} = \sum_{i=1}^{k} y_i \prod_{j=1,\, j \neq i} \frac{x_j - x}{x_j - x_i} = y_i$$

which is the required secret each share is the pair (x_i, y_i) x_i is availed to the public but y_i is kept secret for combinations at appropriate time.