Provable, Secure & Democratic Election Model

Ochieng' Daniel Achola[1], Oleche Paul[2] and Oluoch Nyamwala[2]
[1-2]Department of Pure & Applied Mathematics, Maseno University, Kenya
[3]School of Biological & Physical Sciences, Moi University, Kenya
[1]kazidane2003@maseno.ac.ke, [2]poleche@maseno.ac.ke, [3]foluoch2000@yahoo.com

***Abstract*– Democracy is founded on the principle of elections and opinion expression capabilities. Bimber & Leggewie foresaw e-democracy. Voting is an information transfer model. Trust is essential to communication channel but can't be transferred through that channel. New York Times (November, 2008) editorial and computer security experts like Wallach reported malfunctions, vulnerability and vote flipping. Trust in the correct functioning of the electronic voting system is key to democracy. Identification and verification of voters lie in the design to accurately detect fraud and audit elections. Practical implementation on a bulletin board in a secure way is feasible provided certain deficiencies like privacy, identification, verification and tally is addressed using cryptographic techniques. Lampard's one time signature schemes that ensure one man, one vote principle and converted non-interactive proofs via zero knowledge proofs to identify voters with bit commitments for distributed computation after casting votes can be exploited to achieve this objective. In this paper we have achieved real time tabulation of results in associated race. Simulated interfaces are in the appendix section.**

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

I. BACKGROUND INFORMATION

Modern electronic voting can be traced back to the introduction of Direct Recording by Electronics [DRE] with special voting software which strictly denied access to personal computer based connectors. Fulton & Delkab counties in Georgia were the first jurisdiction to use punch cards and computerized tallying machines in their primary elections of 1964. The Election Technology Council assumed that DRE don't feature keyboard or peripherials that could enable an infiltrator to tamper with software code or vote tabulations but Herbert Thompson, in his response to Frequently Asked Questions "Do Electronic Voting Machines Improve the Voting Process" wrote a five line script in Visual Basic that allowed one to go to central tabulator and change any total tally of votes desired leaving no logs. Saltman in his report *"Accuracy, Integrity & Security in Computerized Vote Tallying*" noted that punch cards contributed to inaccuracy and lack of confidence. Rubin & Wallach [4] evaluated an electronic voting machine based on source code and raised concern about their vulnerability after *Unilect*, Voting Machine Manufacturer claimed that it could store up to 10,500 votes but only held 3,005 after losing 4,438 due its full memory. In a separate but almost similar incidence, Black Box Voting, Inc. security experts Hursh & Thomas, [4] hacked a central vote tabulator without leaving any trace and further showed that the voting machines had backdoor allowing software to be modified or installed several months/years before execution. Dr. Edward [4] demonstrated that in less than a minute of physical access to Diebold Electronic Voting Machine or its PCMCIA memory card, an attacker could install a malware chat could steal votes while modifying all records, logs and counters to be in consistent with fraudulent vote counts and even introduce a virus that spread from machine to other machines. All these reports have necessitated Federal Elections Commission to set standards for computerized voting systems that enhances trust in verification, timely and accurate detection of possible election fraud or malfunctions as a means of audit as was reported by New York Times editorial *"That's Pretty Big Glitch"*, October 8, 2008 where a voting machine malfunctioned to flip votes. Auditing election to detect and prevent election fraud calls for cryptographic techniques as an alternative measure for *Provable, Secure & Democratic Election Models*

II. PRELIMINARIES

A) Basic Concepts

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 ciphertext 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*

Definition 2 *Hash function is a computationally efficient function that maps a binary strings of arbitrary lengths to binary strings of some fixed lengths called hash values*

Definition 3 *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* $b \in \{0,1\}$

Definition 4 *Mixnets are cryptographic alternative to anonymous channels with several linked servers that take batch of encrypted votes randomizes it before outputting a batch of permuted messages such that input and output messages are unlink-able. These mixnets in large scale elections has the property of universal verifiability*

Definition 5 *El Gamal scheme encrypted as*

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

III. OBJECTIVES

In this paper, we

(i) Model a provable, secure and verifiable Hybrid democratic election protocol.

(ii) Simulate an electoral process

IV. MODEL

A) Provable Secure Models

Voting system is an information transfer model. Information is uncertain in nature yet voting is deterministic. Trust is essential to communication channel but can't be transferred using that channel. Elections can be secured over networks through cryptographic techniques. Mixnet is one such secure cryptographic election schemes over networks. Using Chaum's, [8] Mixnet to provide anonymity for group of senders who submits encrypted vote. Voting takes long hours and requires stringent correctness with appropriate mixing inputs that might not alter them.

Sako & Killian, [16] partial decryption

$$\text{Given } c = (\alpha, \beta), \quad \text{PartialDec(c)} = (\alpha', \beta')$$

yields

$$\frac{\beta}{\beta'} = \alpha^{x_i}, \ \mathrm{x}_i \text{ the mix server } \lambda$$

the mix server proves that $g, y, \alpha, \frac{\beta}{\beta'}$ forms a DDH

Decision Diffie – Hellman problem implying that

$$\log_g(y) = \log_\alpha \left(\frac{\beta}{\beta'}\right) \bmod \mathrm{p}$$

For batch proof of knowledge of randomization values, we consider a permutation π and a randomization r_j used by a given mix server to generate another permutation ϕ and lists the new randomization values (t_j). Performing re-encryption and shuffling according to the new parameters to yield secondary shuffle outputs. The verifying Election Authority VA_j challenges the mix server to reveal either (ϕ, t_j) which proves that the second mixing was done by

$$\phi \text{ o } \pi, \mathrm{r}_j - t_j$$

Wakaha et al, [20] noted that if not more than half of the mix-server aborts, they can be excluded from the anomization process dynamically to continue mixing by use of re-encrypt-decrypt mixnet design and adding homomorphic cryptosystem. For a fault tolerance, we employ secret sharing of the decryption server keys. We independently verify Mix servers independent of their numbers, by Shuffling or Decryption as shown by the work of Abe, [1] Mix servers perform secondary mixing to reveal secondary randomization values and permutations simultaneously using commitment scheme. The verifying Election Authority VA_j further challenges the difference between primary and secondary mixes to obtain the difference in sequence. With a modified threshold El Gamal decryption, shareholders coordinate their action to reduce verifiers work. The Mix servers with a quorum build up and in turn decryption factors for each El Gamal mixnet outputs

$$c_{i,j} = (\alpha_j, \beta_j)$$

The Mixnet (M_1) produces

$$\gamma_{j,1} = \alpha_j^{x_1, L_1}$$

Where x_1 a private is key and L_1 is the Lagrange interpolation factor. The Mixnet M_i then produces

$$\gamma_{1,2} = \gamma_j, 1.\alpha_j^{x_1, L_2} \ldots$$

Until a quorum of mix servers effectively produce $\gamma_j = \alpha^x$ which can be used to decrypt $c_{i,j}$ By using Schnorr identification protocol on inputting re-randomization value given two El Gamal ciphertext

$$c = (\alpha, \beta) \text{ and } \mathrm{c'} = \mathrm{RE(c', \gamma)} = \alpha', \beta'$$

Also for a random z

$$G = gy^z \text{ and } \mathrm{Y} = \left(\frac{\alpha'}{\alpha}\right)\left(\frac{\beta'}{\beta}\right)$$

we note that $\gamma = G^y$ anyone who knows γ can perform Schnorr Signature using the public key (ξ, γ). Considering a single mix server M_i with inputs (α_j, β_j) that outputs (α'_j, β'_j) and with properties of $(\mathrm{m \ x \ n})$ permutation matrix $A_{j,j'}$ with elements Z_q we have

$$P_{j,j'} = \sum_{i=1}^{n} A_{i,j} A_{i,j'}$$

Effectively the dot products of the column j and j' and considering

$$P_{j,j',j"} = \sum_{i=1}^{n} A_{i,j} A_{i,j'} A_{i,j"}$$

A_{ij} is a permutation matrix iff

$$P_{jj"} = \begin{cases} 1, & \text{if: j=j'} \\ 0, & \text{otherwise} \end{cases}$$

$$P_{jj'j"} = \begin{cases} 1, & \text{if: j=j'=j"} \\ 0, & \text{otherwise} \end{cases}$$

And by Furukawa-Saka, the proof decomposes the action of re-encrypting mix server as

$$(\alpha_{j'}, \beta_{j'}) = \left(g^{r_{\pi(j)}} \prod_{i=0}^{N-1} \alpha^{A_{i,j'}}, y^{r_j} \prod_{i=0}^{N-1} \beta_i^{A_{i,j}} \right)$$

With Neff's, [] fastest, fully private universally verifiable Mixnet shuffle proof in decomposing mix servers M_i with inputs *j* and some permutation π

$$(\alpha_{j'}, \beta_{j'}) = \left(g^{r_\pi} \alpha_j, y^{s\pi(j)} \beta_j \right)$$

The correct shuffling value occurs when r = s. By letting a vector $(T_i) = \log_g T_i$ of k elements and another vector $(U_i) = \log_g U_i$ all elements of q-order subgroup of Z_q with a chosen generator *g*. Given public input $(T_i)(U_i)r$ and corresponding private inputs $(t_i, u_i), \gamma$ and a permutation π and which proves that $U_i = T_i \pi$ and given a challenge $\omega \in Z_q$ from the verifier, the protocol calls equal exponents on 2k public inputs and letting $U_i = \gamma t_i \pi$ run equal exponents effectively demonstrate that

$$\gamma^k \prod_{i=0}^{k-1} (t_i + \omega) = \prod_{i=0}^{k=1} (u_i + \omega\gamma)$$

This effectively evaluates two polynomials at random point ω. If the two evaluations are equal with overwhelming probability, then the polynomials are equal and there exist a permutation π such that

$$U_i = \gamma t_i \pi$$

If the mix server outputs (α'_j, β'_j) the prover and the verifying Authority engages the protocol to generate a random vector T_j and a random value $r \in < g >$ such that the prover demonstrate the knowledge of

$$R = \sum_{j=1}^{N} r_j, t_j$$

By Jacobson's, [] practical mix and because of flash mixing reliability on repeated robustness. Considering an El Gamal

$$c = (\alpha, \beta) = \left(g^r, m, \gamma^r \right)$$

Blinded by exponent δ to yield

$$\left(\alpha^\delta, \beta^\delta \right) = \left(g^{r\delta}, m^\delta, \gamma^{r\delta} \right)$$

The mix server safely decrypts resulting ciphertext without shuffling as the outputs are blinded and jointly cooperates to un-blind the resulting plain text while shuffling and by repeated robustness a final proof of correct exponentiation by zero knowledge proof. Further usage of El Gamal re - encryption as a fall back whenever error is detected and checking correctness by using layers of encryption as universally verifiable fast Mixnet for honest players. We encrypt the first message as

$$c = \varepsilon(m, r) = (\alpha, \beta) = \left(g^r, m, \gamma^r \right)$$

And hashing *c* cryptographically all elements encrypted in the second layer

$$\phi = (d_1, d_2, d_3) = (\varepsilon(\alpha, r), \varepsilon(\beta, s), \varepsilon(H(\alpha, \beta), t))$$

However a significant blow to optimized technique of repeated robustness and double enveloping like malicious mix server and cancels the effects of mixing. This results in the production of equation of both inputs and outputs rather than knowledge of specific permutations. For semantic security of intermediate ciphertext, a delayed decision is recommended.

B) Our Model

Let $A_1, \cdots A_n$ be n election authorities and $v_1, \ldots v_m$ be *M* voters participating in a multi-hierarchical election. Assuming no collusion of fewer than t authorities can reveal an individual vote $t(1, n)$. Incorporating a digital time signature σ a mechanism that postpones decision on what to vote until the election is completed. We construct ballots as follows:

C) Ballot Construction

Each voter v_i encrypts his vote $b_i \in \{-1, 1\}$ Voter chooses b_i randomly from {-1, 1} and computes the ballot

$$B_i = g^{\alpha_i} h^{b_i}$$

is randomly chosen from Z_q. Voter computes proof B_i to determine the polynomials G_i and H_i

$$G_i(x) = \alpha_i + \alpha_{i,1}(x) + \ldots + \alpha_{i,t-1}(x)^{t-1}$$
$$H_i(x) = \beta_i + \beta_{i,1}(x) + \ldots + \beta_{i,t-1}(x)^{t-1}$$

For these coefficients, voters computes commitments $B_{i,1} = g^{\alpha_{i,1}} h^{b_{i,1}}$ and posts B_i (Proof of $B_{i,1}, \ldots, B_{i,t-1}$) to bulletin board. All participants verify whether ballot β_i is

correctly formed by checking proof β_i Voter sends respective shares $(a_{i,j}, b_{i,j}) = (G_i(j), H_i(j))$ to authority A_j using a private channel. Each Authority checks received share $(a_{i,j}, b_{i,j})$ to verify that

$$g^{\alpha_i} h^{\beta_j} = B_i \prod_{i=1}^{t-1} B^j{}_{i,1}$$

We use Lampard's one-time digital signature scheme to ensure that voters only vote once, hence avoiding multiple or duplication of votes

C) Voter Identification

A reliable identity management system is critical component in applications that render services to only legitimate enrolled users with an overarching task of verifying an individual's identity. Surrogate representations like passwords (knowledge based systems) and identity cards mechanism (token based systems) can be lost, shared or stolen thereby undermining the intended security. Even though biometrics offers natural and reliable solutions to certain aspects of identity management by utilizing fully/semi-automated schemes that recognize individuals based on their inherent physical cum behavioral characteristics. The advent of digital processing and improved sensing technology coupled with significant advances in statistical pattern recognition that can extract salient features sets and compares it with those stored in the data base but still has shortcomings arising from construction of fake fingerprint from gummy clone, Trojan horse scanners that can extract features stored data. In this paper we have proposed a hybrid (dual) identification incorporating both cryptographic techniques like one way hash functions, digital signatures, and protocols like interactive proof systems that can be converted to non-interactive proof systems via zero knowledge proofs. We approximate ridge pattern in local area of the finger using cosine wave

$$w(x, y) = A\cos[2\pi f_o(x\cos\theta + y\sin\theta)] \quad (1)$$

Where $\begin{cases} \text{A}, & \text{Amplitude} \\ f_0, & \text{Frequency} \\ \theta, & \text{Orientation of cosine wave} \end{cases}$

We compute the 2D Fourier Transform of cosine wave. Letting (u, v) denote the location of maximum magnitude then, cosine

wave parameters are $\begin{cases} \hat{\text{A}} = \ |w(\hat{u}, \hat{v})| \\ \hat{\Theta} = \arctan\dfrac{\hat{u}}{\hat{v}} \\ \hat{f}_0 = \sqrt{\hat{u}^2 + \hat{v}^2} \end{cases}$

and since the ridge pattern is not exact cosine wave, we use Fast Fourier Transform (FFT) to compute its 2D Fourier Transform which contains a pair of blurred impulses that can be smoothen the orientation of the fingerprints using low pass filters like Gaussian filters in the interval of $[0, \pi]$. Using Poincare Indexing Method we compute cumulative change of orientation of any neighbour of a pixel

$$P.I = \frac{1}{\pi}\delta\left(O[(i+1)_{\text{mod}8}] - O[i]\right) \quad (2)$$

Where $\delta(\theta) = \begin{cases} \theta\pi, & \text{if } \theta > \dfrac{\pi}{2} \\ \theta, & \text{if } -\dfrac{\pi}{2} \le \theta \le \dfrac{\pi}{2} \\ \theta + \pi, & \text{if } \theta < \dfrac{\pi}{2} \end{cases}$

Poincare Index of a pixel corresponding to a singular point can either be 0-Non Singular, 1-Loop, 1-Delta and 2-Whorl (Combination of two adjacent loops). By assigning the direction of singular points by rotating reference orientation field as

$$RO_{loop}(x, y) = \frac{1}{2}\arctan\left(\frac{x}{y}\right) + \alpha \quad (3)$$

$$RO_{delta}(x, y) = -\frac{1}{2}\arctan\left(\frac{x}{y}\right) + \alpha \quad (4)$$

With □ as the angle in polar coordinates and particularly rotating the orientation field by □ we obtain the following orientation fields

$$RO_{loop}(x, y, \alpha) = \frac{\theta - \alpha}{2} + \alpha \quad (5)$$

$$RO_{delta}(x, y, \alpha) = -\frac{\theta + \alpha}{2} + \alpha \quad (6)$$

D) Verification

Verification of the signature $\sigma = (b, e)$ of the signed message m, σ is achieved by computing

$$c_i := b_j^2 \prod_{i=1}^{t} x_i^{e_j}, \qquad 1 \le j \le k$$

and accepts iff $e = h(m \,|\, c_1 \,|, \ldots |\, c_t \,|)$

Remark 1 *Key size t|n| and signature size k (t+|n|) is secure in random oracle model under the assumption that the hash function is truly random function*

E) Vote Casting

Because of the zero knowledge interactive proofs of bit commitments and in particular electronic voting distributed computation, usage of holomorphic commitment scheme is appropriate. Assuming n voters $v_1, \ldots v_n$ and only that yes/no are possible with a trusted center T computing the election

outcome. For a dishonest trusted center T^* it can determine each voters vote. Letting E_T and D_T be El Gamal encryption and decryption functions for trusted center T. Voter v_i chooses $m \in (0,1)$ at random $r_i \in \{0,\ldots,q-1\}$ and computes $c := Com(r_i, m)$ which is broadcasted to the public while sending $E_T(g^{r_i})$ to the trusted center T that decrypts with the following algorithm

$$D_T(\prod_{i=1}^{n} E_T(g^{r_i})) = \prod_{i=1}^{n} g^{r_i} = g^r, \quad r = \sum_{i=1}^{n} r_i$$

Everybody can publicly compute result s from known commitment

c_i, i=1,...,n and g^r

$$v^s = g^{-r} \prod_{i=1}^{n} c_i \quad \text{with } s := \sum_{i=1}^{n} m_i$$

Therefore a voter v_i simply posts $s_i \in \{-1,1\}$ such that $v_i = b_i s_i$ which is the designated vote. A bit string specific to a voter v_i is included in the input to a hash function H in the proof of the ballot B_i for prevention of vote duplication.

F) Vote Tallying

(i) Each authority A_j post the sum

$$S_j = \sum_{i=1}^{m} a_{i,j} s_i$$

and the sub tally

$$T_j = \sum_{i=1}^{m} b_{i,j} s_i$$

(ii)Each tallier checks the share (S_j, T_j) by A_j to verify that

$$g^{S_i} h^{T_j} = \prod_{i=1}^{m} \left(B_i \prod_{i=1}^{t-1} B_{i,j}^{j'} \right)^{s_i}$$

From t pairs (S_j, T_j) that corresponds to authority for which the shares (S_j, T_j) are correct. Each tallier computes a final tally T

$$T = \sum_{j=A} T_j \prod_{i \in A} \frac{1}{i-j}$$

A is a set of t correct Authorities. We assume without loss of generality that each voter v_i is accepted by all authorities in a successful election.

Remark 2 *In case an Authority A_j receives a share that does not pass the verification step, he can repost it so that any voter can verify it.*

F) Invalidating Authority

Assuming the Election Authority leaves the Election Management Board either through desertion, joining a political party or the key being stolen. The delegation structure shouldn't be indefinitely valid. The invalidation of certification/authority could be done as follows

(i) We include expiry date as part of the delegation

$$\text{Cert } D_{A_i} \rightarrow_a : \text{Sign } \gamma A_i(a_\xi, \text{date})$$

Where

$$\text{Cert } D_{A_i} = \text{Delagating Authority}$$
$$a = \text{Election Official PK}$$
$$\gamma A_i = \text{Delegating Authority SK}$$
$$a_\xi = \text{Delegated Official PK}$$

(ii) If the Delegated Election Official *a* leaves the Election Management Board, his Authority for public key ξ is immediately invalidated (revoked). Let DA_i be the Delegating Authority with serial numbers. We let the delegation structure to be of the form

$$\text{CertDA}_i \rightarrow a := \text{Sign}_{\gamma, DA_i}(a_\xi, \#\#SN\#)$$

where

$$\text{CertDA}_i = \text{Delegating Authority}$$
$$a_\xi = \text{Election Official PK}$$
$$\#\#SN\# = \text{Serial Number}$$

Moreover if the delegated election Official a_i secret key γ corresponding to its public key ξ is stolen, (s)he can alert the Delegating Authority DA_i who searches the data base to find the serial number associated with the certification to create a revocation list that could be eventually posted on a Delegating Authority's webpage.

G) Simulations

Wampserver is a windows web development environment that allows creations of web applications with Hypertext Pre-Processor (PHP) a scripting language that doesn't require a compiler. It's simply maintained with Apache and a Structured Query language (MySQL), the most popular online database. PHP, Apache and MySQL run seamlessly to create dynamic web page alongside PhpMyAdmin that allows easy management of database and web-site automation. Election simulation is designed to engage voters with issues they

pertinent about, give them an opportunity to contest as candidates through political parties affiliations or as independents, run campaigns and vote in an election. On Election Day election management set up polling stations, identify streams depending on the number of registered voters in electronic poll book. Organize security at each of the polling stations within the electoral jurisdiction. Ensure polling stations have sufficient polling booths that allow privacy. Create mechanism to ensure that political parties and their candidates don't campaign within the polling stations on the Election Day. Have Observers, Lobby Groups, Civil Societies and Scrutineers in the elections to countercheck transparency. Poll clerks are responsible for verification of the voters' identity. In this paper, we have proposed an electronic biometric voter verification ID to capture live biometric image and smartcard to verify the eligibility of persons previously registered. Eligible voters are given ballots. Shuffled electronic ballots that include the names and photos of candidates, political party affiliations logo or independent whenever a candidate is not affiliated to any party but excluding titles and prefixes. Some of the interfaces from the simulation of our hybrid cryptographic model are in the appendix section.

V. CONCLUSION

This paper has clarified the requirement and key elements of e-voting system with focus on accuracy of voters' identity and tally using hybrid identification for our voters that incorporates both biometrics and cryptographic techniques to validate eligibility of the registered voters. Identity of voters ensures that there is no violation of one person one vote principle, fitting the principles proposed by IDEA. One time signature on the smartcard with computer chips store information and perform computations internally with physically protected keys and internal protocols ensures security.

Cryptographic techniques used authenticates integrity of data in transit during communication between voting terminals that collect election configuration information , reporting final election results and prevents man in the middle attacks. Our clearly shuffled electronic designed ballots allow voters to mandatorily verify their votes before casting, verify casting time and can be printed, reduces the high frequency of making unintended choices by the voters and spoilt votes. Moreover, shuffled ballots ensure that no candidate has advantage of being at the top of the list. Audio and braille enhancements ease the usability for those with disability conditions and ensure secrecy, incoercibility, integrity and uniqueness of the system. We have achieved real time display of results for the public via bulletin board in pie charts. The Election management has a reliable back up that can be audited for query by a disgruntled party. We have also developed criteria for invalidating certification/authority through expiry of contracts or revocations that can be posted on Delegation authority or elections management web page whenever election officials leave for whatever reason. In this paper we have accomplished a three level voting authority protocol with real time tallying of election results. The first level being county council, the second level being constituency and finally the national level corresponding to councilor, Member of Parliament and presidential seats respectively. We recommend that future research be taken to a four level voting authority protocol with real time tally considering a third level with multiple races. This should include county assembly ward in level one, constituency in level two, county in level three and the entire nation in level four corresponding to member of county assembly, member of parliament, senator, governor, women representative and finally the president respectively taking note that the governor, senator and women representative all belong to the county or level three.

REFERENCES

[1]. Abe Mayasuki: Universally Verifiable Mixnet with Verification Work Independent of Number of Mix Servers *Advances in Cryptology - EUROCRYPT '98*, Springer-Verlag pp. LNCS 1403:446-447, (1998).

[2]. Andrew Neff: Verifiable Secret Shuffle and its Application to E voting *ACM: Conference on Computer & Computer Security*, pp. 116-125, (2001).

[3]. Anil J., Sail P., Matoni D., and Dario M: *Handbook of Fingerprint Recognition* Springer-Varlag, New York (2003).

[4]. Bev Harris: Black Box Voting *Ballot Tampering in the Twenty 21st Century* Bookstore & Library Edition, Talion Publishing. USA (2004).

[5]. Clive Thompson: Can You Count On the Voting Machine New *York Magazine* (January 6, 2008)

[6]. Craig Lambert: Voting into Vapor *Harvard Magazine* November - December 2004, Vol. 107, No. 2.

[7]. Cramer R., Gennaro R., & Shoenmakers B.: A Secure & Optimally Efficient Multi-Authority Election Scheme *Advances in Cryptology- EUROCRYPT '97*, Springer-Verlag LNCS 1233: pp. 103–118, (1997).

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

[9]. Ellein Theison: Myth Breakers: Facts About Electronic Voting

[10]. Fiat, A & Shamir, A.: How to Prove Yourself (Practical Solutions to Identifications & Signature Problems) *Advances in Cryptology-CRYPTO '86* Springer-Verlag LNCS 263: 186-194, (1987).

[11]. Fujioka A., Okamoto T. & Ohta K.: A Practical Voting Scheme for Large Scale Elections. *AUSCRYPT '92* Springer-Verlag LNCS 718 : 224 – 251, (1993).

[12]. http://www.essvote.com/products/12/12 universal voting system.....March 3, 2017

[13]. Markus Jacobson.: A Practical Mix *Advances in Cryptology-EUROCRYPT '97*, Springer - Verlag LNCS 1233:103-118 (1997).

[14]. Rebecca Mercuri: Corrupted Polling, Inside Risks *Journal of Communications* ACM 36(11): 168 (1993).

[15]. Rebecca Mercuri: Verification of Electronic Ballot Systems *Secure Electronic Voting* Advances in Information Security, Vol. 7, Kluwer Academic Publishers.

[16]. Sako K. & Killian J.: A secure & Optimally Efficient Multi-Authority Election Scheme *Advances in Cryptology - EUROCRYPT '95*, Springer - Verlag LNCS 921: 393-403, (1995).

[17]. Shamos Michael: Unilect Corporation PATRIOT Voting System: An Evaluation, April 2005.

[18]. Steve Prettyman: Learn PHP 7*Object Oriented Modular Programming using HTML5, CSS3, JavaScript, XML, JSON and MySQL* Springer Science + Business Media New York (2016).

[19]. www.coe.int/dgap/democracy/act /ggis/evoting, E-voting 202010 Biennial November Meeting.

[20]. Wakaha et al., Fault Tolerance Anonymous Channels *Information & Communications Security'9*, Springer-Verlag ICICS 1334: 440-444, (1997).
[21]. Whitefield Diffie & Martin E. Hellman: New Directions in Cryptography *IEEE. Trans, Inform. Theory*, IT-22:644-654, Nov 1976.

APPENDIX

Simulations

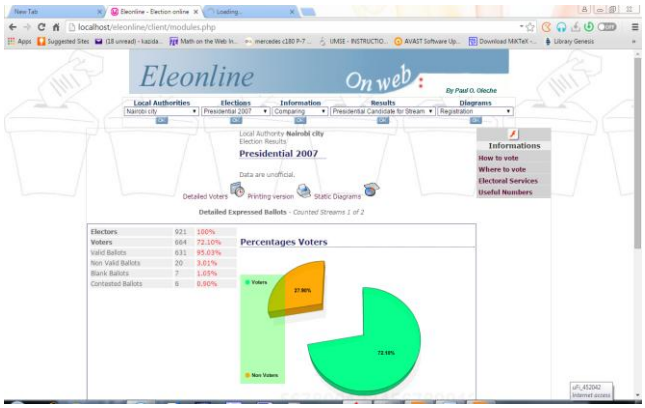

Fig. 1: Presidential Valid Ballots

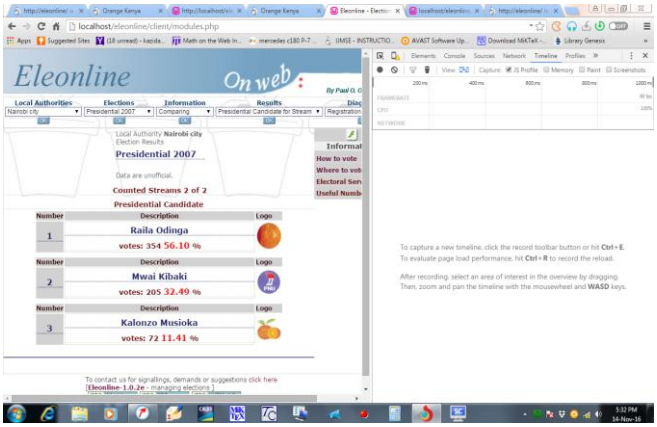

Fig. 2: Presidential Timeline Monitor

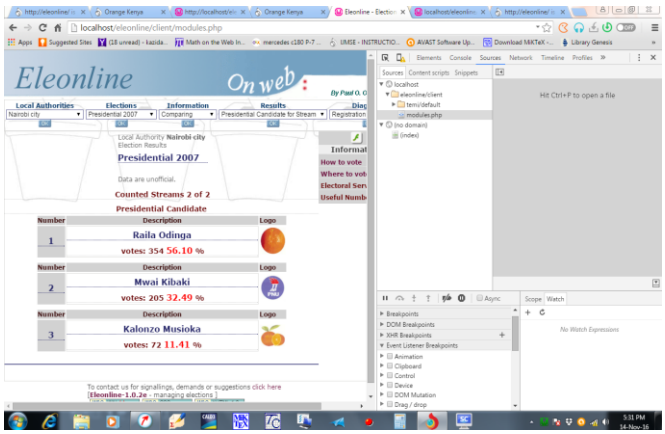

Fig. 3: Presidential Watch

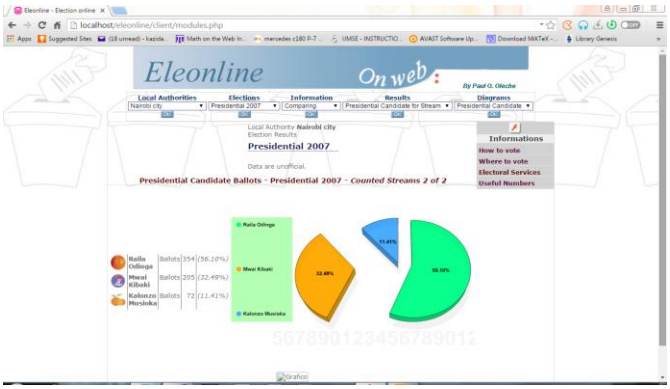

Fig. 4: Presidential Public Bulletin

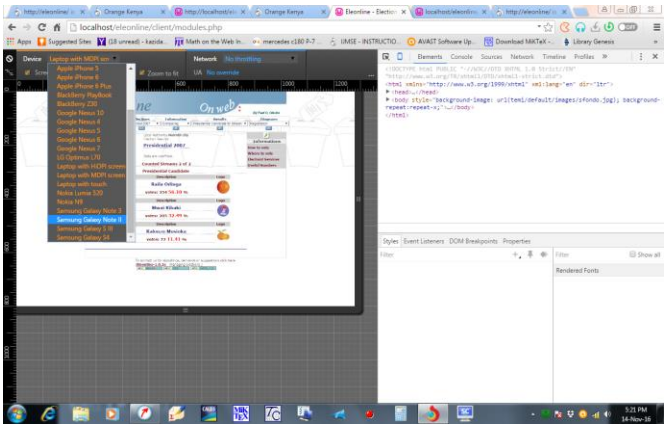

Fig. 5: Network Selection

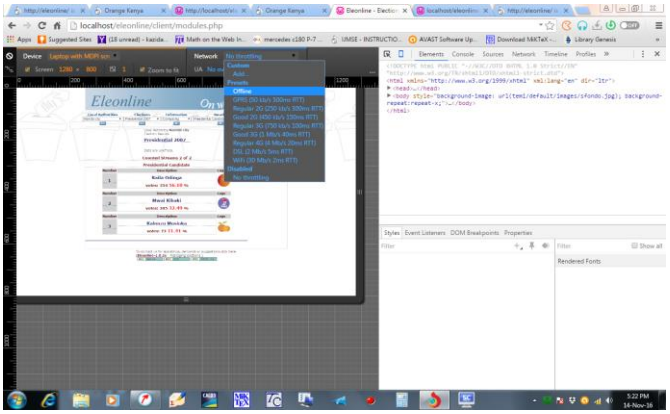

Fig. 6: Device Selection