A biometric-secure e-voting system for election processes

Conference Paper · June 2008

DOI: 10.1109/ISMA.2008.4648818 · Source: IEEE Xplore

CITATIONS
27

READS
20,436

6 authors, including:

Mohammed A Khasawneh
University of Jordan
50 PUBLICATIONS **235** CITATIONS
SEE PROFILE

Mohammad Malkawi
Jordan University of Science and Technology
42 PUBLICATIONS **427** CITATIONS
SEE PROFILE

Omar Al-Jarrah
Jordan University of Science and Technology
69 PUBLICATIONS **1,163** CITATIONS
SEE PROFILE

Thaier Hayajneh
Fordham University
83 PUBLICATIONS **1,689** CITATIONS
SEE PROFILE

A Biometric-Secure e-Voting System for Election Processes

Mohammed Khasawneh[1], *IEEE Senior Member*, *College of Engineering*, mkha@ieee.org,
Mohammad Malkawi, *Vice President, AIM Wireless, USA*, mmalkawi@aimws.com,
Omar Al-Jarrah[2], *Associate Professor of Computer Eng. & IT Director,* **Laith Barakat**[2],
Thaier S. Hayajneh[3], and **Munzer S. Ebaid**[4], *Director of Applied Research*

[1]University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA
[2]Jordan University of Science & Technology, Irbid, Jordan
[3]*School of Information Systems*, University of Pittsburgh, Pittsburgh, PA, 15260, USA
[4]King Abdullah II Design & Development Bureau, Amman, Jordan

Abstract

In this paper we propose a multifaceted online e-voting system. The proposed system is capable of handling electronic ballots with multiple scopes at the same time, e.g., presidential, municipal, parliamentary, amongst others. The system caters for integrity of an election process in terms of the functional and non-functional requirements. The functional requirements embedded in the design of the proposed system warrant well-secured identification and authentication processes for the voter through the use of combined simple biometrics. The design of the system guarantees that no votes in favor of a given candidate are lost, due to improper tallying of the voting counts, with the proper incorporation of system FLAG's. Transparency of voting follows through in all phases of an election process to assure the voter that his/her vote went in favor of his/her candidate of choice. Besides its main functional properties, the proposed system is designed to cater for several essential non-functional requirements. Of utmost importance are the requirements for correctness, robustness, coherence, consistency, and security. To verify the robustness and reliability of the proposed system, intensive computer simulations were run under varying voting environments, *viz.* voter density, voter inter-arrival times, introduced acts of malice, etc. Results of the simulations show that security and performance of the system are according to expectations. These results provide the proper grounds that would guide the decision maker in customizing the proposed system to fit his particular voting needs.

I. Introduction:

In a manual, paper-based election, the electorates cast their votes to select their candidates, where they simply deposit their designated ballots in sealed boxes distributed across the electoral circuits around a given country. By the end of the election period, all these boxes are officially opened and votes counted manually in the presence of certified representatives of all the candidates until the numbers are compiled. This process warrants transparency at vote casting time as well as at counting time.

Often times, however, counting errors take place, and in some cases, voters find ways to vote more than once, introducing irregularities in the final count results, which could, in rare cases, require a repeat of the election process altogether! Moreover, in some countries, purposely introduced manipulations of the electoral votes take place to distort the results of an election in favor of certain candidates. Here, all such mishaps can be avoided with a properly scrutinized election process; but when the electoral votes are too large, errors can still occur. Quite often international monitoring bodies are required to monitor elections in certain countries.

This naturally calls for a fully automated online computerized election process. In addition to overcoming commonly encountered election pitfalls, electoral vote counts are done in real time that by the end of elections day, the results are automatically out [1, 2]. The election process can be easily enhanced with various features based on the demand and requirements of different countries around the world.

Due to worldwide advancements in computer and telecommunication technologies and the underlying infrastructures, online voting or e-Voting is no longer a North American or Western phenomenon. This high tech method of casting a ballot has spread far beyond the **United States**, expanding throughout the entire world. E-Voting, along with its benefits and mishaps, can now be found from the developed countries of Europe to the developing countries of Asia and South America. The introduction of electronic voting has been the biggest change to the Irish electoral system since the establishment of the state over 80 years ago. E-Voting may soon become a global reality or a global nightmare [3 - 5]. Besides

978-1-4244-2034-6/08/$25.00 ©2008 IEEE.

reliable e-Voting technologies, there is a dire need for international standards to govern the technology, the software reliability and accuracy, the processes and algorithms deployed within the technology, and the verification of all hardware, software and protocols involved. Such standards will eventually allow elections to proceed in any part of the world without the need for monitoring bodies.

II. Authenticity of the Voting Process and Privacy of the Voter Rights

Certain factors play out big in a given voting process in any particular country. Culture itself and the underpinning social factors/values largely determine the rules and regulations that govern any voting process. In countries, where election results are determined through the voter counts that are tallied by directly depositing specially designed voting cards into the voting boxes, there are tendencies that electoral votes can get misappropriated in many ways; some voters would tend to attempt to vote more than the number of times permissible by law for a given candidate; other voters may try to vote in lieu of other illegible voters so that the voter count would weigh favorably towards one candidate or another, to mention just a few. Counterfeit/Malice is yet another issue that can jeopardize the integrity of an election process. Automating an election process, while relying on state-of-the-art in computer and ICT technologies, can significantly mitigate many of the factors that would hamper a healthy progress of an election process. Nonetheless, relying totally on available information technologies can only warrant the authentication/validation of the identity of a given voter, but, still, would not have the capacity to block any attempted abuse of the voting system, *viz.,* those voters who simply try to vote on behalf of others (fraud). Without additional measures, the integrity of a voting process, within the proper context, is far from any acceptable standard/s; the incorporation of *biometrics* would definitely have an added value towards achieving the required levels of election integrity.

Present day applications, including banking applications, guarding of high-security establishments, monitoring of passengers across border posts, amongst many others are witnessing increasing levels in the use of biometric technologies and devices. Biometrics is best defined as measurable physiological and / or biological characteristics that can be utilized to verify the identity of an individual. They include fingerprints, retinal and iris scanning, hand geometry, voice patterns, facial recognition, Gait recognition, DNA and other techniques. They are of interest in any area where it is important to verify the true identity of an individual. Initially, these techniques were employed primarily in specialist high security applications; however, we are now seeing their uses and proposed uses in a much broader range of public facing situations.

Essentially, a biometric system follows two characteristic traits: identification and verification. The former involves identifying a person from all biometric measurements collected in a database. The question that this process seeks to answer is: "who is this?" It, therefore, involves a one-compared-to-many match. Verification involves authenticating a person's claimed identity from his/her previously enrolled pattern. "Is this who he claims to be?" is the question that this process seeks to answer. This involves a one-to-one match [6, 7].

Verifying the identity of a person against a given biometric measure involves five phases that the system needs to go through. At the beginning, input data is read from the person through the reading sensors. Collected data is, then, sent across a network to some central database hosting a biometric system. The system will, then, perform identity matching using standardized and/or custom matching techniques. Figure 1 illustrates data flow in a typical biometric identification process.

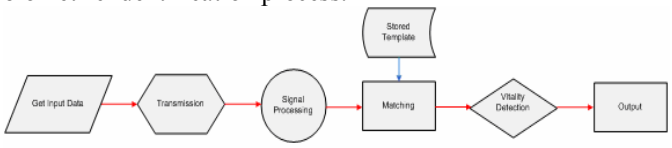

Figure 1 - Biometric System data flow

The incorporation of biometric technologies can be as simple as using a single biometric. However, a single biometric measure is always subject to security breaches, if not properly attended and administered. This naturally includes security passwords, fingerprints, and signatures, all of which can be spoofed when applied in a non-properly attended environment. This is significantly alleviated and system security enhanced with the proper application of combined simple biometric measures. The application of combined weak biometrics leads to systems that are less complex and more robust in terms of the security levels attained. There are strong single biometric measures which involve retinal and iris scans that are rather hard, if not impossible, to breach, but usually lead to more complex systems which, in turn, slow down the underlying biometric matching process due to the amount of data processing involved. For these reasons, amongst others, the type of biometrics addressed in this work is of the former type that involves combined biometrics of the weak types. This will be elaborated upon in the succeeding sections.

Section III of this paper provides a description of the proposed e-Voting system. This system was initially proposed for parliamentary elections in Jordan[1]. Section IV presents the simulation model used to evaluate the proposed system, together with results and findings. Section V addresses system susceptibility and issues in cyber security. Conclusions are given in section VI.

III. The Proposed *e-Voting* System:

In this paper, we propose client/server web-enabled e-Voting software architecture. The architecture is illustrated in Figures 2a and 2b shown right across.

Besides the main functional properties of a voting system, as described in the previous section, the eVoting system must cater for several essential non-functional requirements. Of utmost importance are the requirements for correctness, robustness, coherence, consistency, and security.

On the server side, a global database is maintained for all registered voters and candidates. Also, the server runs in real-time and provides backend statistics for the entire election process.

On the client side, two more requirements are necessary. In order to reduce the traffic rate on the network links, a local database at the client side is required to host the data which pertains to the local voting center. This DB is a rather dynamic one, in the sense that the data stored in its tables may vary over the election time period. The size of the local DB at any voting center is only a small fraction of the global DB at the server side. The use of a local DB enhances the performance of the voting process. However, this approach creates a synchronization problem, which will be addressed later in this section.

The second requirement is the transparency of the voting process. In essence, a voter at an electronic voting station casts his/her vote to a computer. The voter does not have an insight on how his/her vote is translated and/or counted. In a paper-based election, the ballot is filled out by the voter and dropped into a sealed box by the voter himself/herself. Votes are counted in the presence of candidates or their representatives. The voter is certain that his/her cast ballot with his/her vote selection is in the right box. Of course, ambiguity in the ballot formats (as was the case in the US presidential election in 2000) may render the transparency a rather deceiving one. In an electronic version, the voter puts his trust into computer hardware, software and network infrastructure that processes his/her vote. Hence, the e-Voting system in its broadest form may render the process a non-transparent one [3, 8].

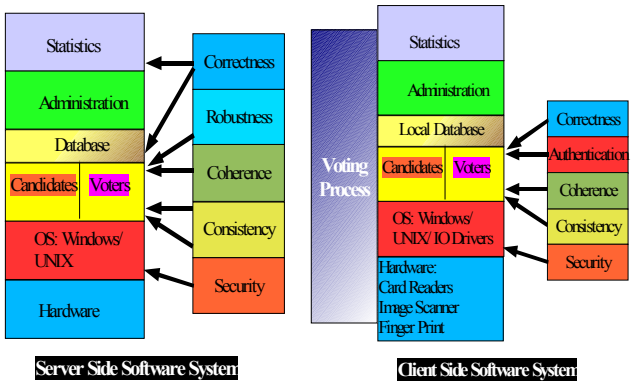

Figure 2: a) Server b) Client

We propose a two-sided solution to the transparency problem. On the one side, the system prints a hardcopy of the vote cast by the voter. The voter verifies the accuracy of his/her vote and retains the copy for his/her records. On the other side, the system generates another copy of the vote with a new unique key identifier; the name and identity of the voter is concealed. This copy is saved in a secure box and can be used later to verify the correctness of the votes as stored in the final DB destination. This side of the copy can be printed out as a bar code which can be easily scanned and read automatically. Only a randomly selected set of these copies need to be tested. This two sided process guarantees transparency by providing verification of the accuracy of how the cast vote is input into the system and then how it is, finally, stored in the DB tables.

One of the challenges facing an e-Voting system is to insure that no voter can impersonate another voter and no voter can vote more than one time. In the proposed system, we use an identification followed by an authentication process. The identification is done via a card reader which reads the official ID card of a voter and pulls the voter record from the local DB or loads the record from the central DB if it is not found in the local one. The voter record includes a biometric description of the voter. In this study, we use a fingerprint authentication method. The voter will be rejected if his/her fingerprints do not match the stored ones. In order to reduce false rejections, we store for each voter several copies of his/her fingerprints taken at different time intervals. Fingerprints are stored as an encoded text in order to reduce storage consumed by images. This dual process should guarantee that no one can falsely impersonate a voter.

In order to prevent two or more votes per voter, we use a "voting status flag" in the voter record. This flag is initialized to FALSE. The voting status flag is set to TRUE in the central DB whenever a voter identity is verified (before authentication takes place). If the authentication fails, the flag is reset to FALSE. If the voter leaves the station without completing a vote, the flag is also reset to FALSE; thus allowing the voter another chance to try

[1] The project was funded by King Abdullah II Fund for Development (KAFD), grant # 11/2006, and sponsored by King Abdullah II Design & Development Bureau (KADDB)

again to cast his/her vote. If the voter completes the voting process, the flag remains set to TRUE. Note that even if the result of the vote is not committed to the central DB in due time, the flag in the voter's central record is set to TRUE, thus eliminating the possibility of another attempted voting by the same voter, or by someone who carries a counterfeit ID card. This requires that whenever the record of a voter is accessed for identification, even when the record is found at the local DB, the flag on the central record must be checked. If it has already been set to TRUE, the voter is denied access and his/her attempt fails. If two people carrying the same ID card (one is real while the other is counterfeit) attempt to vote at the same time, the first one to access the record will set the flag to TRUE, load the record and prevent the other one from accessing the record. Of course if the one with the counterfeit card obtains the record, the vote cast will fail at the next authentication step. It is possible that a record gets loaded into two different voting centers due to block transfer from the central DB into local DB's. When a voter attempts to access the record at any of the stations, the client will verify the central record flag. If it has been set to TRUE, access is denied; otherwise it sets the flag to TRUE and access is granted. Note that simultaneous requests to the same record will be synchronized by the DB query serialization process (only one query may access any table at any give time). This mandatory check of the flag in the central DB, however, will add extra overhead on the network. This overhead will be further evaluated in the simulator, but will not be reported in this study due to time and space constraints.

Another synchronization resolution is required when a vote is to be registered in the record of a candidate. If a candidate is being selected by several voters at the same time, then a certain assignment plan needs to be placed in order so that all votes will be tallied (no misses) and added to the candidate's record. Again we use a "count" flag/mutex for the candidate's record. The COUNT flag is set initially to FALSE. When the record is selected by a voter, the flag is set to TRUE until the record count is updated, then the flag is reset to FALSE. All votes for the same candidate will be queued until the flag is reset to FALSE. A copy of the vote will be printed only when the vote is successful and the candidate's record is updated. This requirement, initially made for transparency purposes, provides a final test for the accuracy and correctness of the process, especially in the presence of thread hangups. The correctness and accuracy of the system using the two flag attributes is demonstrated (physically present) in the current simulation study. When the flags were turned OFF, we noticed several violations and accuracy problems. Those were remedied when the flags' attributes were turned ON.

The voting process, as discussed above, is shown in the flow diagram of Figure 3. The overall architecture of the system is shown in Figure 4. The central database, Figure 4, which is mirrored out for reliability reasons, is used to store all relevant information on the candidates and voters. Voting centers are distributed around the country. One or more voting centers could share a local database. At a voting center, each voting station is equipped with a card reader, a fingerprint scanner, a touch screen, and a multimedia subsystem. The multimedia subsystem is used for people with special needs (physically challenged), such as the blind and those with difficulties in reading or comprehending images, texts, or sounds.

The proposed system is capable of handling electronic ballots with multiple scopes at the same time, e.g. presidential, municipal, parliamentary, and others. However, the simulation environment in this study is designed only for a single voting scope.

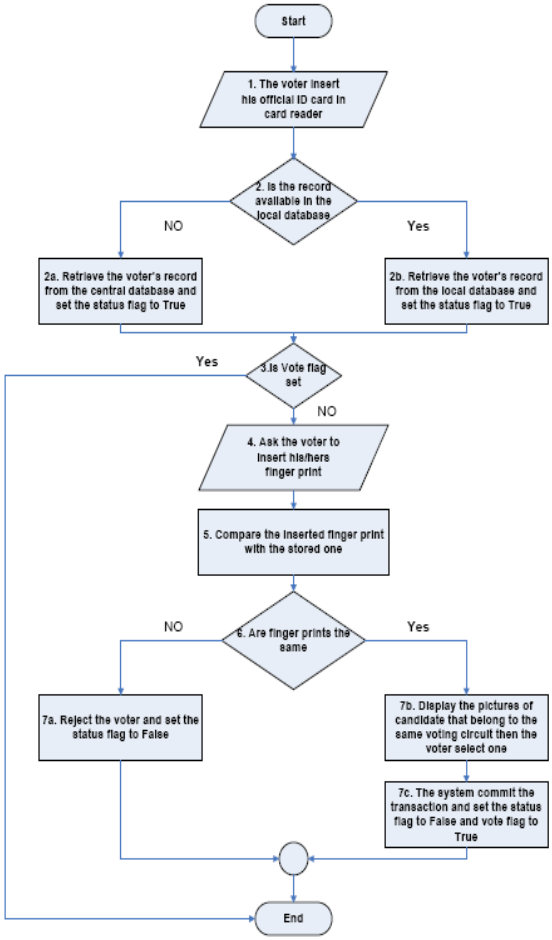

Figure 3- Voting Process Flow Chart

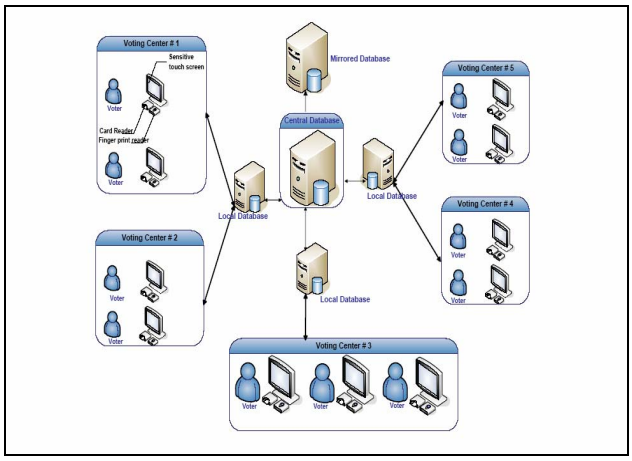

Figure 4- General schematic diagram of proposed system architecture

IV. Simulation Results:

A simulation model has been built in order to test and evaluate the behavior of the proposed electronic voting system. The simulation is, also, useful for providing proper guidance on configuring the eVoting system in terms of server requirements, network bandwidth, voting stations, and the like.

The simulation environment includes an Oracle database system for voters and candidates. Besides personal identification information, the records include authentication information and locality of a voter and/or a candidate. The simulator, also, includes modules which emulate the arrival of voters at voting centers and the voting process itself. The simulator allows a voter to cast a vote at any voting center, irrespective of his actual voting district (locality). This is one of the main advantages of e-Voting systems.

Voters arrive at a voting station according to a Poisson arrival process, and the temporal distance separating the various arrivals is modeled as an exponential random variable. The hypothetical maximum number of voters arriving at a voting center is set by the system admin *a priori*; this is explained by the fact that the number of voters in a given voting district is known beforehand. Each voter would swipe his/her official identification card through a magnetic card reader, at which point he/she would be prompted for his/her finger print upon completion of which a candidate screen would pop up showing pictures of candidates in the electoral circuit of the voter. If the voter's record indicates other needed forms of display/presentation (as embedded in the information on the voter's ID card), such as sound, then those forms will be used instead of the candidate image display/s. The voter would select his/her candidate of choice at the touch of an image displaying the picture of his/her candidate of choice. The system also allows the voter to cast the vote via audio means for those voters with special needs. At this point the voting process for a given voter is complete and the voter count is tallied in favor of the chosen candidate.

In the simulator, the speed of the voting process is governed by a number of limiting factors: First, a growing queue length was seen to adversely impact the rate at which voters were able to cast their votes. Second, the response time of the system, right from the minute a voter would step into a voting center until the cast vote is tallied in favor of one candidate or another, is adversely impacted by the database response at the server end. Third, the network response time, *viz.*, available network bandwidth, plays out big at determining the transaction time per voter. In our simulations, and for the particular purpose of this paper, we have assumed that the network bandwidth is infinite. We will investigate the network impact on the voting process in an ongoing study. However, using the client/server model with the embedded local DB infrastructure, we anticipate minimal impact of network constraints on the overall process.

Although we have conducted a fairly large number of simulations of the proposed voting system, taking the number of voters over a sample range starting at 5000 voters per voting center and ending at 20,000 voters per voting center, and due to space limitations of this publication, we restrict our assessment of the model to 5000 voters per voting center as our case study. The total number of voters at a given center is fairly constant, since it depends mostly on people who reside in the vicinity of a voting center. So we chose to fix the number of voters at a given voting station in the simulator. In reality, this number may vary by a small percentage due to the fact that people will be allowed to vote at any other center they choose for the sake of voting convenience, especially those voters residing at townships outside their voting districts, or those voters casting their votes through embassies outside their home country.

The other parameters that affect the outcome of the simulation are the number of voters (voter density) arriving simultaneously at a given station and the average time between two successive arrivals. We model the first parameter as a Poisson random process with an average arrival rate (λ). The inter-arrival time is modeled as an Exponential random process with an average inter-arrival time (μ). Note that the system will be rather stressed for large λ and moderate μ (Figure 5). The queue length can grow indefinitely, and the voters will wait in-line for ever. Note that this result is obtained from voters casting their votes at one voting station. One way of resolving this problem is by adding one or more stations, where the voters are split equally between the stations. In essence, each station will receive voters at a rate of λ=5 instead of 10. Keeping the same inter-arrival rate, we observe that the system becomes stable and the queue length and waiting

time are fairly finite (Figure 6). Note that in the simulations, we did not include the human-related response time, e.g. walking, typing, and figuring out what to do, and so on. So the average waiting time in the figures reflects a system-constrained waiting time due to queuing activities. The results, as shown, are averaged over three computer runs to mitigate any bias in the simulation outcomes.

It is well known that on Elections Day, the voting turnout varies over time. During the early hours, the turnout is usually low, then it picks up around mid-day, then it slows down. Occasionally, bursts of voters arrive heavily towards the end of the voting period. Figure 7 shows the system behavior for low voter turnout with λ=2 and μ=2.

Figure 8 illustrates the case where the number of voters arrive in small numbers, however the flow of voters is rather fast (λ=2 and μ=10). The average queue length is 15, and the average waiting time is 3.2 seconds.

Note that it is possible to control the queue length and the average waiting time in a queue. When the voters burst volume, i.e., voter density is large, adding one or more voting stations will relax the problem. When the burst rate is high, i.e., voters arrive at a faster rate, queuing up the voters in line outside the voting center will alleviate the pressure on the electronic voting system. Note that the voters will continue to experience long waiting times; however, the system response time will continue to be acceptable and indefinite postponement or starvation can be avoided.

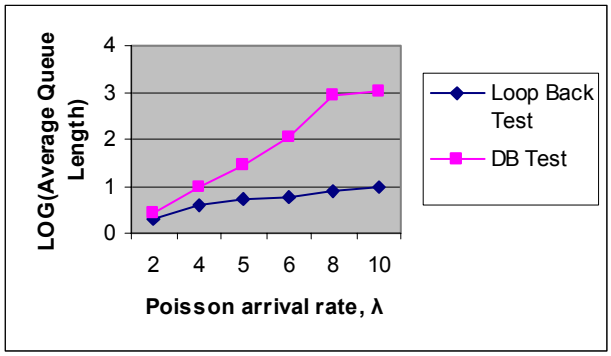

Figure 9-Log Average Queue Length vs. Arrival Rate Parameter (λ)

We finally analyzed the various simulations we obtained to verify the ruggedness/susceptibility of our proposed e-voting system. The plot in Figure 9 compares the average queue length vs. the voter arrival rate at a given voting station. From the figure it is noted that the DB response, since we were running a live DB in our simulations, plays a rather important role in determining the growth of the queue length; where the queue length remained at acceptable levels when we emulated the situation of ZERO DB response time (Loop Back test), including the live DB access (DB Test) showed that the queue length started to grow exponentially.

V. System Susceptibility and Issues in Cyber Security:

Information security is very important to our system. There is inherent need to secure all the communications between the clients and their local DB servers. We also need to secure the communications between the local DB(s) and the central DB server. The system may use the Internet or any other public network to connect the local servers or the clients. Thus the system is vulnerable to many attacks such as [9]:

- Interruption, delay, denial of receipt or denial of service; in such cases the assets and information are made unavailable.
- Interception or snooping; in this case, an unauthorized party will be able, by browsing through files, eavesdropping, or reading communications, to gain access to private/sensitive information.
- Modification or alteration; in this case, information in transit is changed or stored for later access by an unauthorized party.
- Fabrication, masquerade, or spoofing; in this case, an attacker may inject spurious information into the system and make it look like it had originated from a legitimate entity.
- Repudiation of origin; this is a fake denial that an entity did (send/create) something.
- There are also other possible attacks as: replay attacks, denial-of-service and session hi-jack.

To achieve security assurance we need to ensure that all the following objectives are met:

- Confidentiality: Keeping data and resources secret or hidden.
- Integrity: Ensuring authorized modifications; includes correctness and trustworthiness. May also refer to: Data integrity and Origin integrity.
- High Availability: Ensuring authorized access to data and resources whenever desired.
- Accountability: Ensuring that an entity's action is traceable uniquely to that entity.
- Non-repudiation: Preventing false denial of an act.

In order to reach the desired level of security in our system we propose the use of Kerberos. Kerberos is a computer network protocol that provides high level of security for parties that communicate over non-secure networks and allows them to communicate in a secure manner. Kerberos was originally proposed by researchers at Massachusetts

Institute of Technology (MIT) [10, 11]. It was used by many systems all over the world and proven to be secure and dependable.

Kerberos will allow mutual authentication for the clients and the servers over insecure connections. It, also, provides protection against eavesdropping and replay attacks. We believe that it will provide the best option to immune our system against all the aforementioned attacks.

Kerberos requires that the clients and the servers be loosely synchronized in time. For the system shown in figure 4 we propose the use of a double Kerberos protocol. The first one will be between the clients and their local DB server. The second one will be between the local DB servers and the central DB. The reason why we propose the use of double Kerberos is to achieve a higher level of security and completely separate the central server from its clients.

With Kerberos each local server will be separated to an authentication server (AS) and a service server (SS); in our case the local DB server. Initially all the clients need to contact the AS to authenticate themselves using a long-term shared secret. Usually, the shared secret or key is derived from the username and password of the user who logged on the client machine. After the AS verifies the identity of the client it will provide it with a ticket. This ticket will be used later by the client to request additional tickets from the AS to the SS (our local DB servers). These tickets can be used to get services from the SS. If the local servers need to contact the central server, then they have to use the second Kerberos protocol. In this case the local servers act as clients to the central sever and the same process is repeated.

One point of concern remains, however, when Kerberos is used. Since Authentication is made relying solely on the Authentication Server, makes of the AS a single point of failure. Once an authentication server is down, communication between the clients and the SS is halted. This results in disabling an ongoing voting process, and, hence, an imminent increase in queue lengths of voters in the sectors affected. To mitigate this effect, a redundant AS can be put in stand-by mode, which can take over in the event of outages to the main AS.

VI. Conclusions:

In this paper, we have proposed an online e-voting system which can tackle all earlier issues encountered in a conventional (manual) voting system. The new system maintains voting statistics in real-time while preserving the integrity of the voting process from the minute a voter steps in to cast his/her vote until the cast vote is registered in favor of the chosen candidate at a globally allocated DB repository. While observing full-fledged voting transparency, at the voter as well as the system levels, the proposed system is capable of denying access to any illegal voter/s, preventing multiple votes by the same voter, and blocking any introduced forms of malice that would adversely affect the voting process altogether. Moreover, the proposed voting system caters for the needs of the physically challenged voters by providing special multimedia amenities that would facilitate voting to a voter's convenience.

While carefully observing the security needs of the system, at all levels in the voting process, the design of the system also caters for a number of important functional and non-functional requirements, which are sufficiently addressed in every facet of system design which entail hardware, software, and the underlying encryption and network infrastructure.

Simulation results of the system, while running a live DB backend server, reveal a number of important factors that ought to be assessed carefully by the party adopting a system like this one, for any form of election activities, prior to its final deployment. These factors address the number of voting stations needed at any voting center, as outlined by the voting needs of a given voting district, the network bandwidth requirement by a given voting center, the size of the local DB to support the needs of a given voting locality, amongst others. The system, via these simulations, has shown ruggedness and sustained reliability in terms of preventing multiple votes by the same voter, and maintaining internal system audits that would warrant no missed votes, per candidate, in the process of voting.

With the use of an e-voting system, as the one proposed in this paper, many of the issues, that have challenged traditional voting systems in the past, are bound to be resolved providing peace of mind to both voters and election candidates. It is well expected that with a well administered/designed e-voting system, countries that have long been observed by international monitoring bodies, while carrying out election processes of their own, will soon be able to work on their own and, yet, achieve the election integrity they have longed for.

References:

[1] R. Mercuri. Electronic Vote Tabulation Checks and Balances. PhD thesis, University of Pennsylvania, Philadelphia, PA, October 2000.

[2] A. D. Rubin. Security considerations for remote electronic voting. Communications of the ACM, 45(12):39–44, December 2002. http://avirubin.com/e-voting.security.html

[3] McGaley Margaret, McCarthy Joe, "Transparency and eVoting: Democratic vs. commercial interests", www.cs.nuim.ie/~mmcgaley/Download/Transparency.pdf

[4] Online Voting. Parliamentary Office of Science and Technology. May 2001. www.parliament.uk/post/pn155.pdf
[5] McGaley, Margaret. "Irish Citizens for Trustworthy Voting." 6 July 2004. http://evoting.cs.may.ie/
[6] Joshua Smith with Advisor Dr. S. Schuckers,Improving Usability and Testing Resilience to Spoofing of Liveness Testing Software for Fingerprint Authentication,2005
[7] S. Nanavati, M. Thieme, R. Nanavati. *Biometrics: Identity Verification in a Networked World.* John Wiley and Sons, Inc. 2002.
[8] TADAYOSHI KOHNO, ADAM STUBBLEFIELD, AVIEL D. RUBIN, DAN S. WALLACH: Analysis of an Electronic Voting System, IEEE Symposium on Security and Privacy 2004.
[9] Introduction to Computer Security, by Matt **Bishop** (ISBN: 0-201-44099-7), Addison-Wesley 2005.
[10] C. Neuman, T. Yu, S. Hartman, K. Raeburn, The Kerberos Network Authentication System (RFC4120), July 2005.
[11] K. Raeburn, Advanced Encryption Standard (AES) Encryption for Kerberos 5 (RFC3962), February 2005.

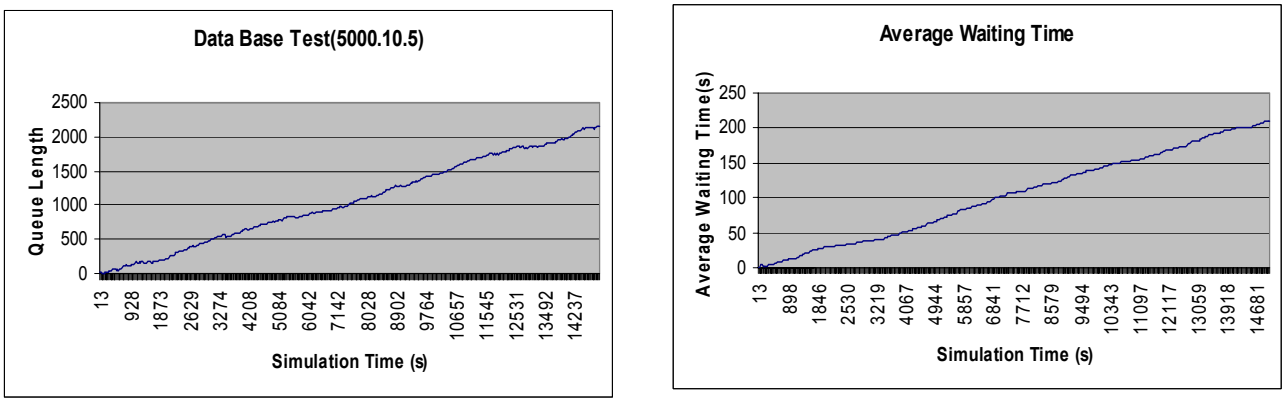

Figure 5: Large arrival rate and moderate inter-arrival time