Implementation and Evaluation of Steganography based Online Voting System

[1]Lauretha Rura, [2]Biju Issac and [1]Manas Kumar Haldar
[1]Swinburne University of Technology (Sarawak Campus), Malaysia
[2]Teesside University, Middlesbrough, UK

Abstract
Though there are online voting systems available, we propose a new and secure steganography based E2E (end-to-end) verifiable online voting system, to tackle the problems in voting process. This research implements a novel approach to online voting by combining visual cryptography with image steganography to enhance system security without degrading system usability and performance. The voting system will also include password hashed-based scheme and threshold decryption scheme. The software is developed on web-based Java EE with the integration of MySQL database server and Glassfish as its application server. We assume that the election server used and the election authorities are trustworthy. A questionnaire survey of 30 representative participants was done to collect data to measure the user acceptance of the software developed through usability testing and user acceptance testing.

Keywords: Online voting system; image steganography; visual cryptography; usability testing;

1. INTRODUCTION

One of the most important concerns in elections is to have an efficient and secure voting procedure. Even though it could be achieved by implementing an e-voting system, its ability to complete voting process faster than the paper ballot procedure alone does not guarantee its security. E-voting systems must be able to earn user's trust and confidence by providing enhanced security features without affecting usability, efficiency and reliability. The system should offer some level of transparency to the user without allowing any breach of trust and privacy. To fulfil this condition, e-voting systems must provide both individual and universal verifiability. Individual verifiability is the ability of an e-voting system to offer vote verifiability to the voter through the implementation of vote receipt, whereas universal verifiability is the ability to offer election transparency to its users. Such systems are categorised under End-to-End (E2E) verifiable voting system (Adida, 2008). End-to-end verifiability represents a change in electronic voting, allowing a way to verify the integrity of the election by permitting the voters to use the system generated information, rather than trusting that the system has behaved correctly (Ryan, Schneider & Teague, 2015). In this paper, we propose an improved E2E verifiable voting system called as eVote software. This voting software could deliver a secure, reliable, convenient, and efficient voting system. As a research objective, we want to improve the quality of election procedure in an electronic voting system that relates to security and usability aspects, by using visual cryptography and image steganography in the system architecture. We also want to evaluate the developed online system through usability testing.

The outcome of evaluating an Internet voting system in the Canton of Zurich shows the need to rely on more advanced technology and centralised infrastructure (Beroggi, 2014). In the work (Azougaghe, Hedabou & Belkasmi, 2015) an electronic voting system based on homomorphic encryption to ensure privacy and confidentiality are proposed. The eVote software differs from previous online voting systems with the usage of cryptography and steganography to secure the data transmission during the election. The difference between cryptography and steganography lies in the way data is processed.

Cryptography generates a ciphertext, while steganography produces a stego-object which is not perceptible by Human Visual System (HVS). In electronic voting, cryptography is a commonly used technique as it is a good defence against threats. In this paper, the authors introduce a novel approach to enhance E2E Voting System's security by combining visual cryptography with image steganography. Image steganography is chosen due to its capability to use data transmitted over the network. During the election voting process, the image steganography protects the existence of the message as a secret (Wang and Wang, 2004), offering a good solution for threats and risks that might occur. The combination of these two schemes is expected to produce an improved and secure approach (Morkel et al., 2005). Petcu & Stoichescu (2015) proposed a mobile biometric-based design that uses techniques such as Secure Sockets Layer encryption, certificate keys and security tokens. This paper is organised as follows. Section 2 discusses the E2E verifiable voting system and related works, section 3 is the proposed eVoting system, section 4 is the software testing and the usability analysis done, and section 5 is the conclusion and limitations.

2. E2E VERIFIABLE VOTING SYSTEM

Various E2E systems have been proposed and are widely used these days (Ryan et al., 2009; Chaum, 2004; Adida, 2008; Chaum et al., 2008; Hubbers, Jacobs, & Pieters, 2005). A verifiable voting system allows blind voters and voters in remote locations to cast fully secret ballots in a verifiable way (Burton, Culnane & Schneider, 2016). In principle, E2E voting system offers assurance to the voters over their cast vote. This is done by distributing vote receipt of encoded cast vote to each of the voters for verification purpose. To support this verification process, E2E systems implemented bulletin board which is a secure append-only broadcast media where each of the encoded votes would be posted once the voters completed the voting process. To verify their cast votes, they need to match the encoded value on their receipt against the values shown on the bulletin board. However, the vote receipt cannot be used as a proof of vote buying or vote coercion because it is encoded. As a result, the E2E voting system would protect the voter's privacy and supports incoercibility that preserves the integrity and impartiality of the election result. This mechanism is illustrated in Figure 1.

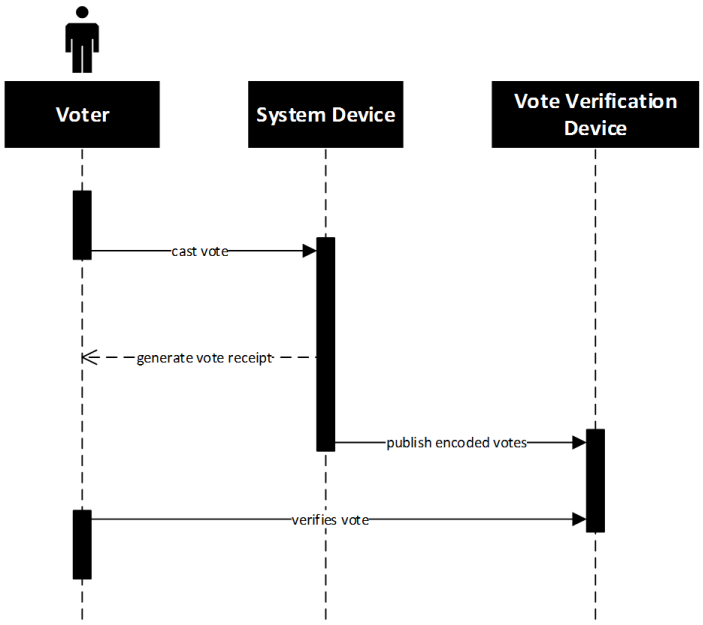

Figure 1. Basic E2E voting system's mechanism

E2E Voting Systems Requirements

Every e-voting system has numerous requirements to be fulfilled to ensure its primary characteristics – individual and universal verifiability is intact. These requirements are mostly categorized as non-functional requirements. Listed as follows are the non-functional requirements of E2E verifiable voting system in general (Fujioka et al., 1992; Benaloh, 2006; Gritzalis, 2002; Cetinkaya, 2008; Kofler et al., 2003; Aditya, 2005):

- Completeness – All valid votes are counted correctly.
- Soundness – A dishonest voter cannot disrupt the voting.
- Privacy – All votes must be secret.
- Un-reusability – No voter can vote twice.
- Eligibility – Only authorised voters are allowed to vote.
- Fairness – Nothing must affect the voting. (i.e. no one can indicate the tally before the votes are counted)
- Verifiability – No one can falsify the result of the voting.
- Robustness – The result reflects all submitted and well-formed ballots correctly, even if some voters and (or) possibly some of the dishonest election officials cheat.
- Incoercibility – It is not possible for anyone but the voters themselves to acquire any information regarding their secret ballots; even if the voters are untrustworthy (the election process is assumed to be conducted by the voter in private).
- Receipt-freeness – Each voter can neither obtain nor be able to construct a receipt to prove the content of their ballot to anyone else.
- Mobility – No restrictions on the location from where a voter can cast a vote.
- Convenience – The system must allow voters to cast their votes quickly, in one session with minimal equipment or special skills without compromising its usability.

Related Works on E2E Voting Systems

E2E voting systems vary based on their security and flexibility levels. In this section, four different types of E2E voting systems that have been used in medium to large-scale real-world elections will be discussed to give a better understanding of E2E voting system. They are: (1) Helios voting system for Recteur election of Université Catholique de Louvain in Belgium, (2) Scantegrity II in Takoma Park municipal election, (3) Student Council election at Princeton University that makes use of Prêt à Voter system and (4) Rijnland Internet Election System (RIES) for public election in Netherlands (Carback et al., 2010).

Helios Voting System is an open-source web-based voting system that offers verifiable online elections (Adida, 2008). It was designed to ensure a clean election setting through the open-audit election, unlike a typical traditional election where only the election officials are entitled to do the observation throughout the election process. Its latest version offers a better approach to protecting system's privacy by appointing multiple trustees, given the main assumption that the trustees will remain truthful. This enhancement was inspired by the simple, verifiable voting protocol proposed by Benaloh, in which they implemented Sako-Killian mix-net scheme and threshold decryption cryptosystem. Each trustee has to decrypt the final tally of the election by using advanced

cryptographic techniques. This open-audit election also ensures universal verifiability. Individual verifiability is done through the implementation of vote (receipt) verification feature called ballot tracking centre where users can verify whether their votes have been received and tallied correctly. This vote receipt is shown to the users in ciphertext format.

Unlike Helios, Scantegrity II increases election integrity through a novel use of confirmation codes printed on ballots in invisible inks (Chaum et al., 2008). It is a practical enhancement for the initial optical scan voting systems – Punchscan and Scantegrity. The physical ballot of Scantegrity II consists of a voting portion and a receipt portion. Just as the traditional paper ballot voting procedure, the voters are given conventional paper ballot where they need to mark their chosen candidate with a special pen that uses invisible ink. This technology allows the voters to retain their receipts in a secure and secret manner with the help of unique confirmation codes on each ballot that no attackers would be able to coerce. The confirmation codes on voter's ballot are kept secret and will only be visible to the voters when they cast their votes. No information regarding the confirmation codes would be accessible to anyone before the votes are cast. Due to this feature, Scantegrity II can earn more trust and confidence of the voters which results in individual verifiability by the voters themselves. Besides that, the system also provides universal verifiability for everyone to reconfirm the computation of the tally and ensures that votes are not altered or deleted for manipulating the final tally of a particular election. Through the implementation of invisible ink in its vote verification feature, Scantegrity II could prevent some of the issues raised by the Punchscan and Scantegrity, like phantom votes and randomization attack.

Besides introducing Scantegrity II and its predecessors, Chaum also published a paper on Secret-Ballot Receipt Election (2004) which inspired Peter Ryan and his fellow researchers to develop Prêt a Voter System (2009). It implemented the same concept as Chaum's secret-ballot receipt scheme with visual cryptography approach proposed by Naor and Shamir (1994) in a simpler way. Prêt a Voter System was introduced to provide more accurate and faster tallying process to cut unnecessary election cost and to increase voter participation. The election auditability feature allows any of the system's users including the audit teams to evaluate its integrity by checking distinct stages of voter authentication, ballot preparation and vote processing. This system supports both universal and individual verifiability. Similar to other E2E voting systems, Prêt à Voter assure the voters that their votes have not been altered. Votes were collected and counted correctly in the tally by giving each of the voters a unique encrypted receipt. This receipt will not leak out the ballot; it can only be used to check the vote status against the read-only bulletin board. With the support of some security components, vote verifiability could be ensured. These security techniques give a better security where internal sources of threats could be anticipated and handled properly. The security components include encryption schemes such as RSA, ElGamal and Paillier and few other cryptographic methods like threshold decryption cryptosystem, zero-knowledge proofs, homomorphic encryption, etc. Thus, the vote would remain secret, and the possibilities of election fraud could be averted.

The last E2E voting system that we want to discuss is RIES (Rijnland Internet Election System). Similar to other E2E voting systems, the RIES was developed to increase the actual number of voters participating and to decrease the unnecessary cost of the conventional election via mail (Hubbers, Jacobs & Pieters 2005). The system was initially designed by Herman Robers for the completion of his master's thesis (Robers 1998). It was then implemented at a local election in the Delft University of Technology. Soon after Hoogheemraadschap van Rijnland, a local water management authority in Netherlands continued its development. RIES allows eligible voters to cast their votes in two distinct techniques - either by mail or electronically. Based on this key feature RIES allows its users to independently verify the election's result. RIES voting system which was implemented in water boards election differs from the initial system by Robers because of the implementation of some features to ensure that it provides internet voting in a simple, straightforward and transparent way without

sacrificing the system's reliability, performance and maintenance cost. Those changes are the elimination of multifunction smartcard to authenticate the voters, which was substituted by digitalized secret key and the supplementary feature of vote by regular mail integration and additional user's type in the system. The use of RIES is abolished as security problems were found in the implementation.

As discussed, all E2E voting systems were designed to fulfil two main objectives, to provide individual-verifiability (also known as voter-verifiability) and universal-verifiability. The four voting systems reviewed are equipped with both features. However, the authors are proposing a novel E2E voting system that is capable of fulfilling all the requirements of the E2E voting system without compromising its integrity, security and usability.

3. THE PROPOSED SYSTEM

The eVote software is an improved version of the existing end to end verifiable voting system. While the existing E2E voting systems cater to different scales of the election, eVote is intended to assist the voting process in small to medium scale election. It not only offers secure and reliable voting system, but it also provides a flexible platform for the election officials to set up and maintain it based on their needs. System users are divided into three distinct types (levels) – voters, polling officers and system administrators. Its technology and system stages are described below.

System Technology

The eVote voting system is built as a web application that can be accessed through a computer or a tablet. Due to its low platform dependency as well as other characteristics such as security, robustness and scalability, Java EE 6 has been chosen for the eVote's system architecture. Security is very important in the development of an E2E Verifiable Voting System. Voting procedures in an online election rely on various information security building blocks that have to do with cryptography. Cryptography is used due to its general defence against electoral frauds like ballot box tampering and other attacks. We also introduce steganography to complement the cryptography schemes. Steganography offers better protection against threats and attacks similar to vote tampering by maintaining secret communication between two parties (client-side and server-side). It is used to protect the data transmitted between the voter and the server to ensure that it would not be accessible to anyone but the voters. Image steganography is used in our proposed system. The various implemented technologies are discussed as follows.

Password Hashed-based Scheme

Password hashed-based scheme is applied to secure user's password in registration and authentication stage. It does not require extensive computation, yet it is proven to be cryptographically secure (Wagner & Goldberg, 2000). Hashed-based algorithms are one-way functions, and the ciphertext form of hashed value is not reversible into the original plaintext. The server is only required to compare the hashed value calculated from user input with the hashed value stored in the database for user authentication. To enhance this protocol and make it even more difficult to be compromised by known attacks (like dictionary and brute force attack on stored pre-computed passwords), salt value and key stretching are implemented alongside its algorithms. To generate a completely secure salt value, reliable Pseudo-Random Number Generator (PRNG) is used. In eVote's development, Java EE SecureRandom Class was used with 24 bytes of salt value. However, even with the enhancement of salt value, the intruder still can steal the user's password by running dictionary or brute-force attacks on each password hashed. Therefore, PBKDF2 key stretching technique is introduced to strengthen the password. In Java EE, SecretKeyFactory Class supports this technique. This class constructs secret keys by using PBKDF2 function found in RSA Laboratories' Public-Key Cryptography Standard

(PKCS) #5 v2.0. The standard name for this secret-key algorithm in Java documentation is PBKDF2WithHmacSHA1.

Visual Cryptography

Visual cryptography (Naor & Shamir, 1994) was implemented to prevent vote buying or selling, as well as vote coercion by providing direct assurance to each of the voters through digital vote receipt. The plaintext (in this case the ballot), will be encrypted to two shares of ciphertext. The ciphertext has two separated layers of pixel symbols. For an additional layer of security of the ciphertext shares, Java EE SecureRandom Class was applied in secret message distribution over the shares. This class produces cryptographically strong random numbers by implementing its Pseudo-Random Number Generator (PRNG) algorithm - SHA1PRNG. It uses the SHA-1 hash function as the foundation of the PRNG. One share will be given to the voter as their vote receipt, and the other share is to be stored in the database. To decrypt these shares, visual cryptography decryption algorithm is to be executed. This mechanism was adopted from Chaum's secret-ballot receipt (Chaum, 2004). A simple amendment is made to the applied mix-net scheme used by Chaum (2004) due to its extensive process.

Threshold Decryption Cryptosystem

Even with the implementation of cryptography based security, attacks and threats still cannot be averted. There are enormous numbers of possible attacks in a remote e-voting system. Besides using visual cryptography and password hashed-based scheme, the eVote software also implemented threshold decryption cryptosystem as an additional layer of security. Shamir developed threshold decryption cryptosystem in 1979. A (k, n) threshold scheme secures and provides reliable key management for a cryptographic system. By having robust security and protection over the key management itself, the security of a cryptographic system itself could be ensured. Threshold scheme would be implemented in the ballot decryption process of the tallying stage to ensure that only authorised personnel can have access to the vote tallying process. To perform this decryption process, the private key, which has been divided and distributed to a few appointed personnel, must be merged before each of the election officials as well as the election administrator can gain access to the summary ballot list which is also known as the 'ballot box'.

Image Steganography

Steganography is the science of hiding information when two parties communicate, where others in between would be unaware of the hidden information. Image steganography provides an enhanced security technique of data encoding with the digital image file as the cover file. Based on our previous work (2011), F5 image steganography algorithm (Westfeld, 2001) as in figure 2, is considered to be more efficient for secure data transmission compared to the other image steganography schemes.

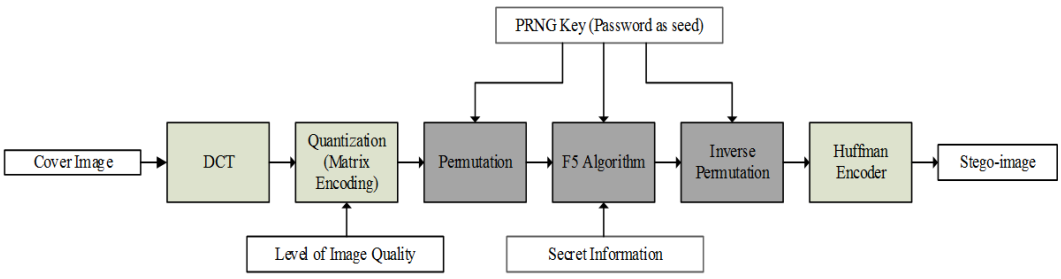

Figure 2. Message encoding process of F5 Steganography Algorithm (Westfeld, 2001)

F5 Steganography technique has better characteristics compared to the other image steganography algorithms namely - LSB, Palette-based and Spread Spectrum. One of the evaluations conducted was the comparison of initial and stego-image sizes for different image steganography techniques as shown in Figure 3 and F5 looks better overall. The F5 stego-image size is small and hence can transmit the embedded stego-image to the election server faster (Rura, Issac & Haldar, 2011). The other comparison is carried out to examine the robustness of each image steganography techniques against visual attack and statistical attacks namely - Regular Singular (RS) analysis and Binary Similarity Measures (BSM) test respectively.

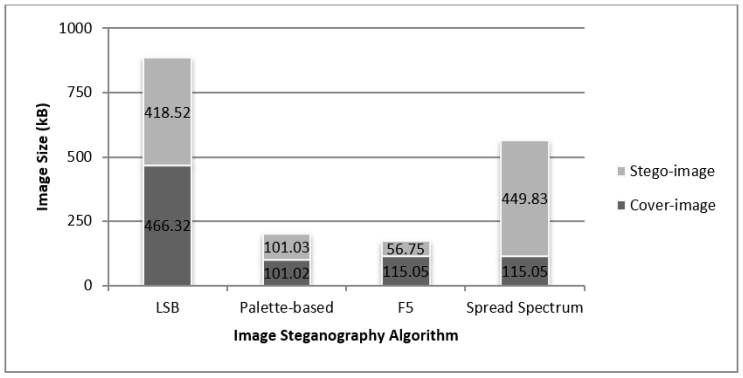

Figure 3. Comparison of initial and stego-image size on different implementation of image steganography techniques

Based on the results displayed in Table I, where robustness of each image steganography methods is ranked from low to high, it can be seen that F5 is not highly subject to visual attack. F5 also eliminates the possibility of Chi-square (χ^2) attack (Bateman, 2008). Considering figure 3 and table 1, F5 image steganography algorithm was chosen in our system. Fridrich, Goljan & Hogea (2002) explained breaking the F5 algorithm in their paper, but the authors Fard, Akbarzadeh-T & Varasteh-A (2006) discusses a new genetic algorithm (GA) approach for secure steganography.

Table I. Robustness of different image steganography methods to visual and statistical attacks

Image Steganography methods	Steganalysis method used	**Visual Attack**	**Statistical Attack**
LSB	RS Analysis	Low	High
Palette-based	BSM Test	Low	High
F5	BSM Test	Medium	High
Spread Spectrum	BSM Test	High	High

The System Stages

The electoral process in the eVote software consisted of five stages – registration, authentication, voting, tallying and vote verification. The process flow diagram of the software is shown in figure 4. Further explanation of each stage of the software is as follows.

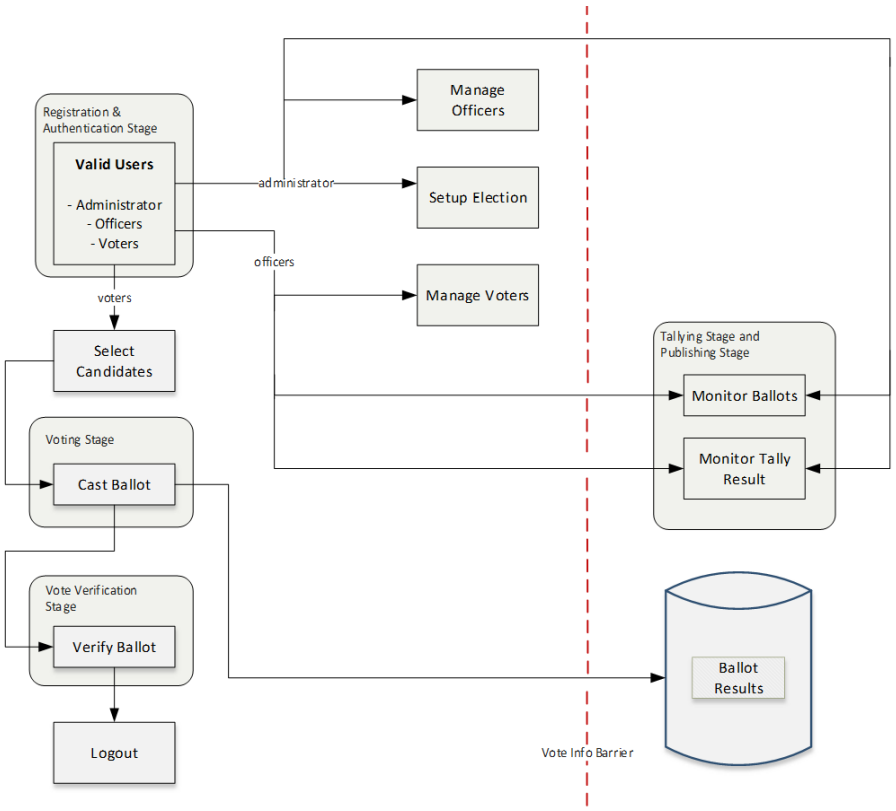

Figure 4. Process flow diagram of the eVote software system

(1) Registration Stage

In this stage, all constraints for the election are prepared by the voters and polling officers. Before this stage, system administrators must prepare the election setup by adding the details of the election and the candidates for each election category. Besides that, the system administrators also need to add the records of eligible voters and polling officers in the database. This record includes their username, Identification Card (IC) number and valid email address. Upon successful attempts, eligible users will receive emails from system administrator notifying their eligibility to register into the eVote system as shown in Figure 5.

By accessing the link provided in the email, eligible voters and officers can now register themselves in the system. To register themselves, the users are required to provide their details and submit. These are then matched with user details saved by the administrators in the database to ensure the accuracy of the details given by the users. As another layer of security, the user's passwords will be cryptographically secured by applying password hashed-based scheme for the generation of salt value and hashed password. By using this, only the hashed password together with its salt value is saved into the database. Its implementation will be explained in the next stage of the eVote software system, which is the authentication stage. There is a slight difference between the registration process of voter and officer. Each voter attempting to register in the system will be randomly assigned to a polling officer. This is done as an additional layer of protection over the database records which will be described further in the tallying stage. After successful registration, users will be directed to their respective homepage by the system.

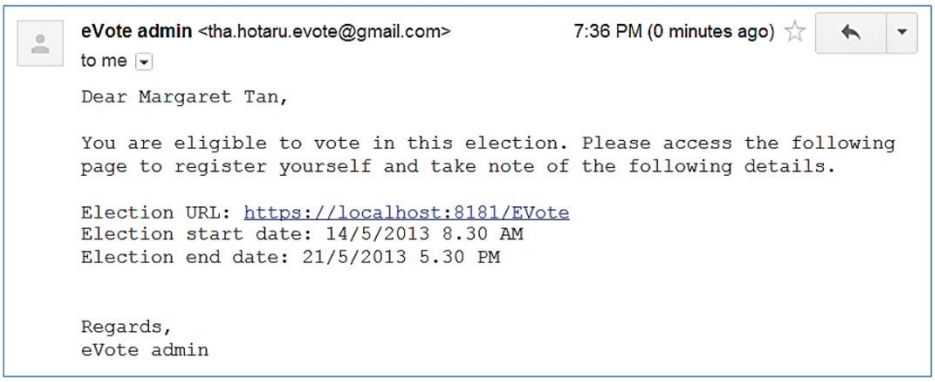

Figure 5. Screenshot of the email received by an eligible voter

(2) Authentication Stage

In a remote e-voting system, the implementation of this stage is mandatory. The objective of this stage is to ensure voter's identity. Registered voters are authenticated by logging into the system. They will be prompted to enter their self-defined username and password for security purpose. The user's passwords are not saved in the database, but only its hashed values. As hash-based algorithms are one-way functions, the hash values cannot be converted back to a plaintext. To authenticate users, the system is required to compare the hashed value calculated from user input with the hashed value stored in the database. Once a user has been identified as a registered voter and has successfully logged into the system, he will see a welcome screen which shows the user account status and a menu panel where a user can navigate through features offered depending on the user level. Figure 6 shows the homepage screenshot for a voter.

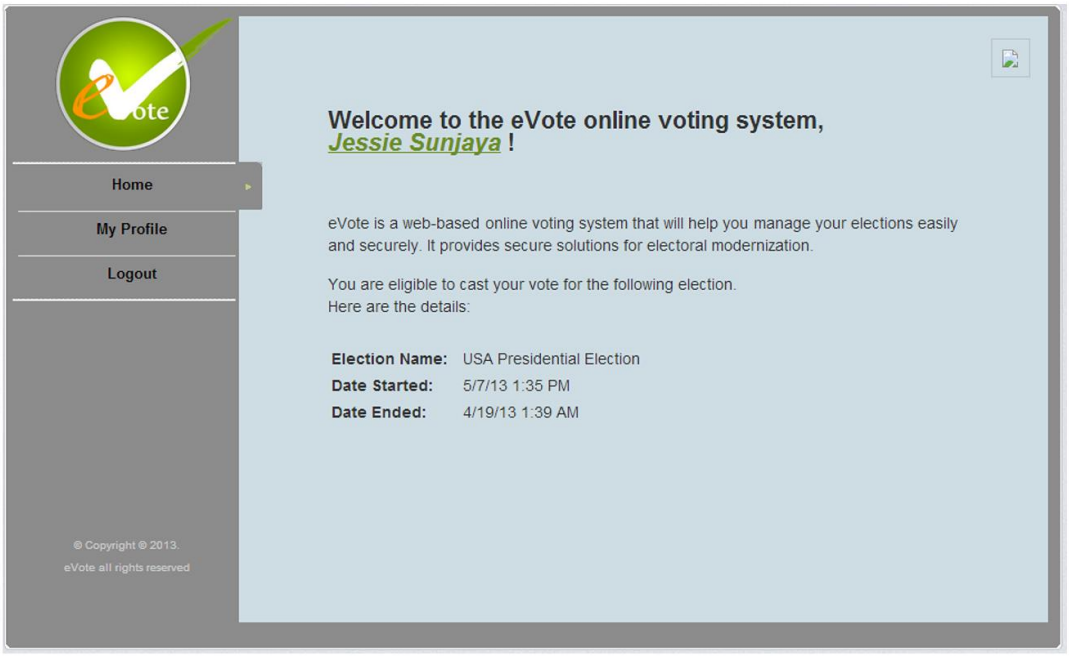

Figure 6. Screenshot of voter homepage upon successful registration

(3) Voting Stage

This stage is carried out by forming a secured ballot electronically and sending it to the election server

where all the ballots would be collected and stored. After completing the two stages mentioned above, voters can then log on to the system and access the voting page. They can cast their vote by selecting their desired candidates for each category listed on that page. The voter's ballot is generated every time the chosen candidates are reviewed or updated. Figure 7 shows the voting page, where the voter can choose one candidate.

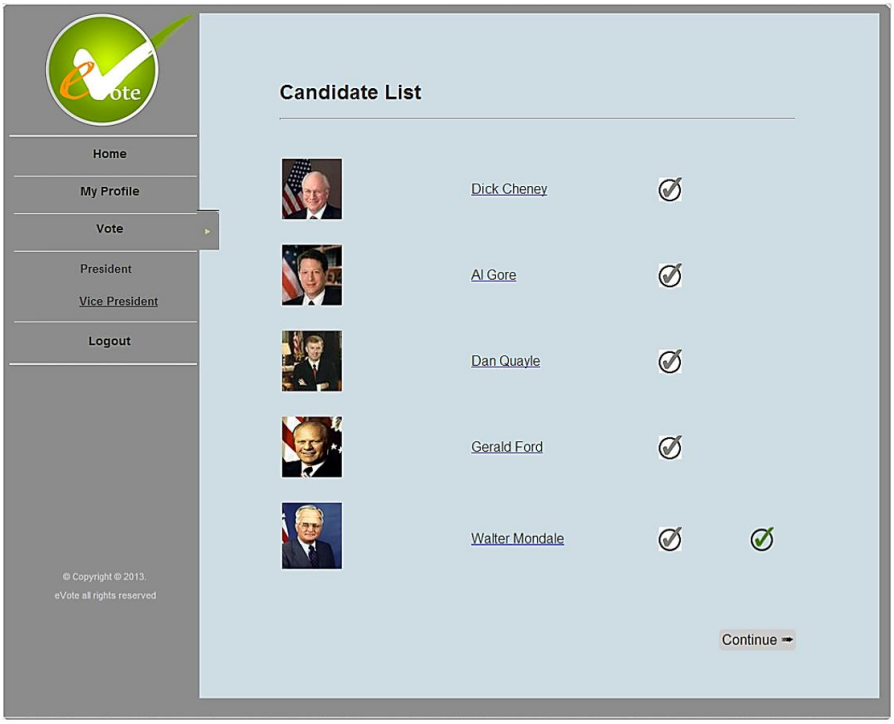

Figure 7. Screenshot of voting page accessible only for the voters

During the ballot generation, F5 image steganography algorithm would be applied as shown in Figure 8. The voter's chosen candidates would be encrypted in a stego-image format for their ballot. This ballot will, later on, be sent over to the tally server.

Input: message, shared secret, cover image
Output: stego-image
initialize PRNG with shared secret
permutate DCT coefficients with PRNG
determine k from image capacity
calculate code word length $n \leftarrow 2k - 1$
while data left to embed **do**
get next k-bit message block
  repeat
  $G \leftarrow \{n$ non-zero AC coefficients$\}$
  $s \leftarrow k$-bit hash f of LSB in G
  $s \leftarrow s \oplus k$-bit message block
    if $s \neq 0$ **then**
      decrement absolute value of DCT coefficient Gs
      insert Gs into stego image
    end if
  until $s = 0$ or $Gs \neq 0$
insert DCT coefficients from G into stego image
end while

Figure 8. Pseudo-code of F5 Image Steganography algorithm applied in the voting stage (Provos & Honeyman, 2003)

Once received by the server, the ballot would be decrypted to reveal the candidate names before they are encrypted again with visual cryptography as an additional security level to earn voter's direct trust by providing the vote receipt. The decrypted stego-image (ballot) would be encrypted with visual cryptography technique by splitting the vote into two shares. The stand-alone share would not reveal any information to anyone, but once the shares are overlaid or combined using a visual cryptography decryption algorithm, the voter's casted vote would be revealed. Basically each voter would be given one layer or share of the image as their receipt which will be sent to their respective email account as displayed in Figure 9, while the other separated layer of the vote would be kept by the administrator for ballot counting purpose and to disconnect the relation of each voter with their own ballot.

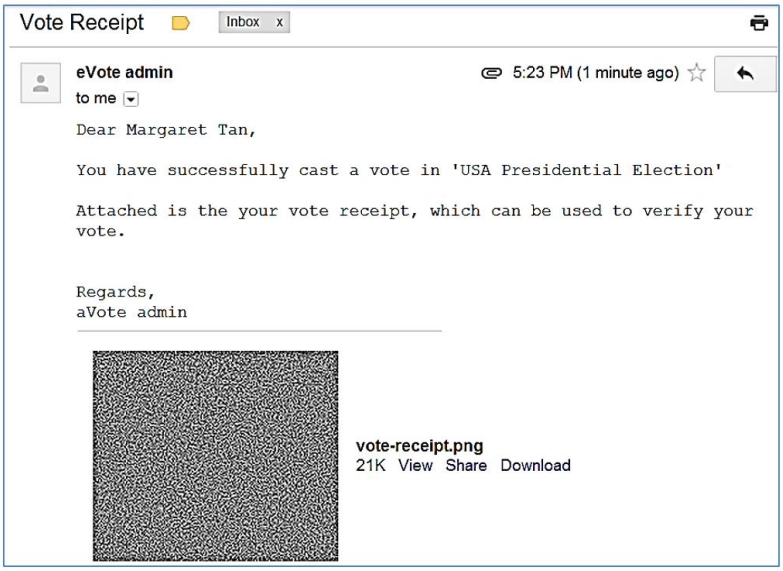

Figure 9. Screenshot of the voter's vote receipt received by the voter

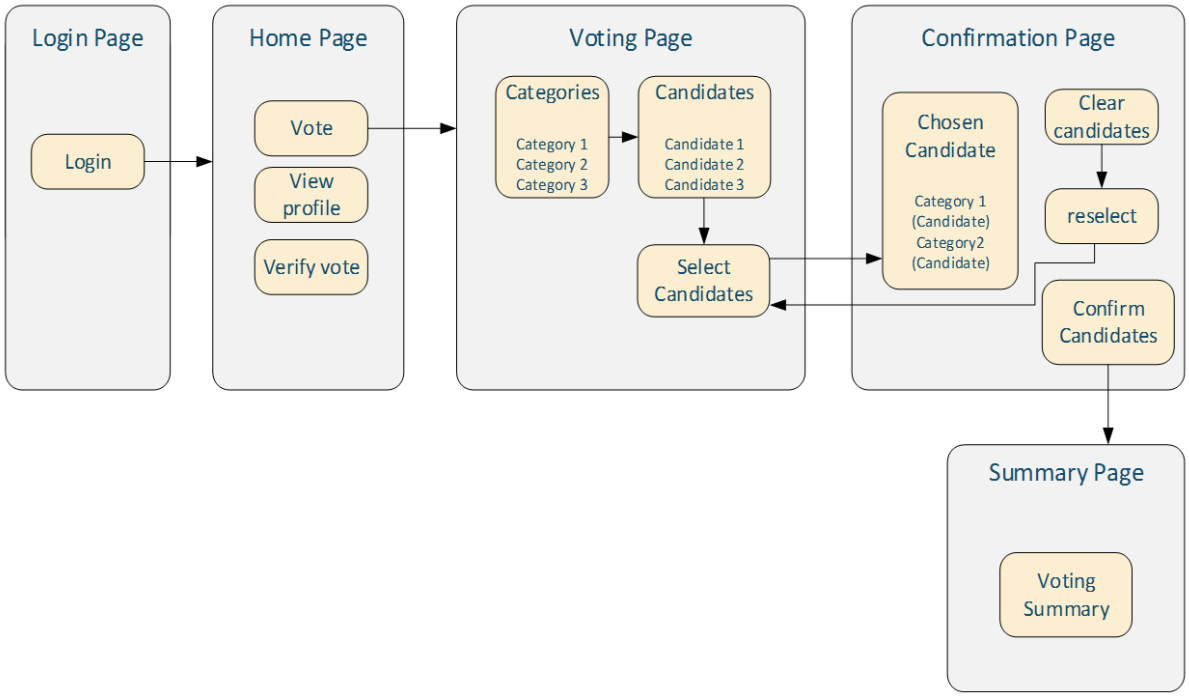

Figure 10. Process flow diagram of voting stage

In the eVote software, the voting stage process is finished when the voting summary page is shown. The overall process flow of the eVote's voting stage is shown in Figure 10.

(4) Tallying Stage

Tallying stage follows the voting stage. After the votes are cast, ballots are securely stored in the database. Users cannot access the ballots before the completion of the tallying stage. The tally determined at this stage is obtained by polling officers with help from system administrators. Each polling officers holds a unique secret key to retrieve ballot records from the database. These keys are pre-distributed by the system administrators during the election setup. System administrators generate these keys by utilising UUID utility.

To access the tally list, polling officers must perform 'decryption' process by merging their secret keys. This method is called the threshold decryption cryptosystem (Shamir, 1979). Only after each of the polling officers has submitted their secret keys, the result tally list (bulletin board) is accessible to the system administrators and the polling officers for monitoring. This tally list is only readable and does not show any relation between the ballot and its voter. Threshold scheme is implemented in the ballots decryption process to ensure that only the authorised personnel can count the vote. The tally is shown in figure 11.

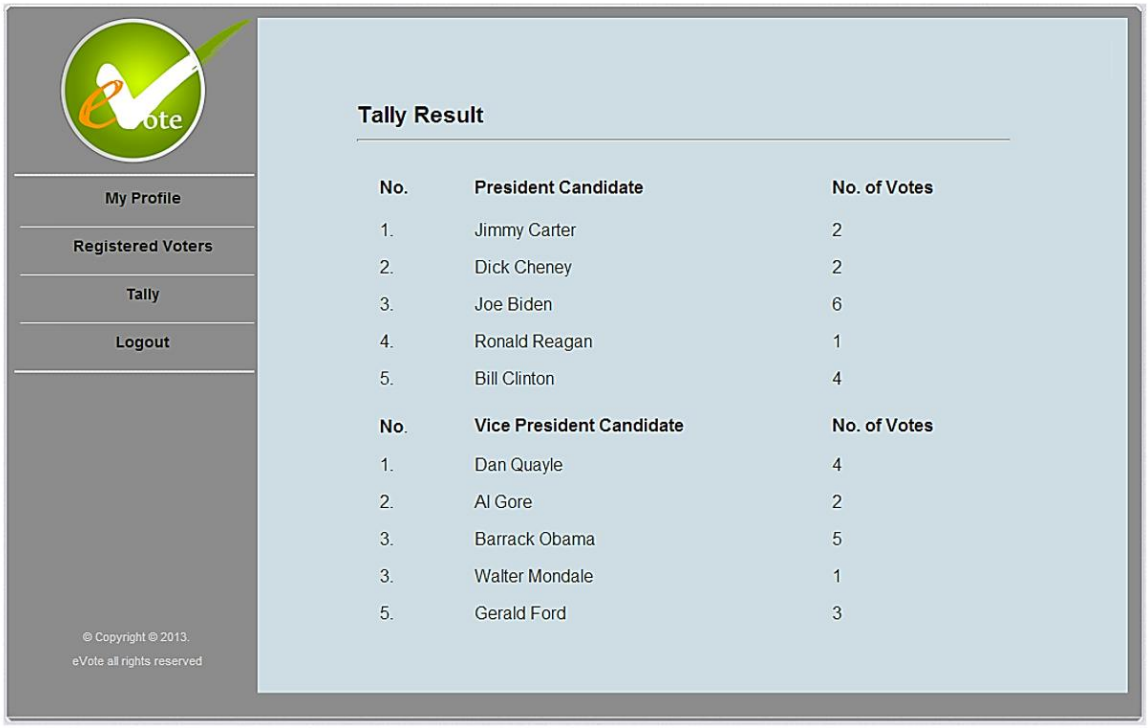

Figure 11. Screenshot of tally results page

(5) Vote Verification

In a traditional paper-based voting, once the tally process is done, authorised personnel will announce the result of the election. However, voters will not be able to verify their votes. As a result, voters cannot be assured that their submitted ballot is counted as cast. This may affect the turnout in subsequent elections. To solve this problem, voter receipt is implemented in the system development of the E2E voting system. This receipt is not revealed in their ballots. It can be used by each voter to

ensure that the ballot cast is properly used by the system. Each user can only obtain one share of the visual cryptography encrypted image. The other half of the shares is automatically saved in the database. The combined shares would be used to retrieve and verify the voter's ballot. Voters of such a system can verify their votes by submitting the vote receipt into the system. The verification feature is supported by visual cryptography scheme. The vote receipt submitted by the voters will be matched (decrypted) against the other half of the encryption share saved in the database during the voting stage to verify an individual's vote. The server storing the share is assumed to be secure. The vote verification is shown in figures 12 and 13 (Rura, Issac & Haldar, 2011).

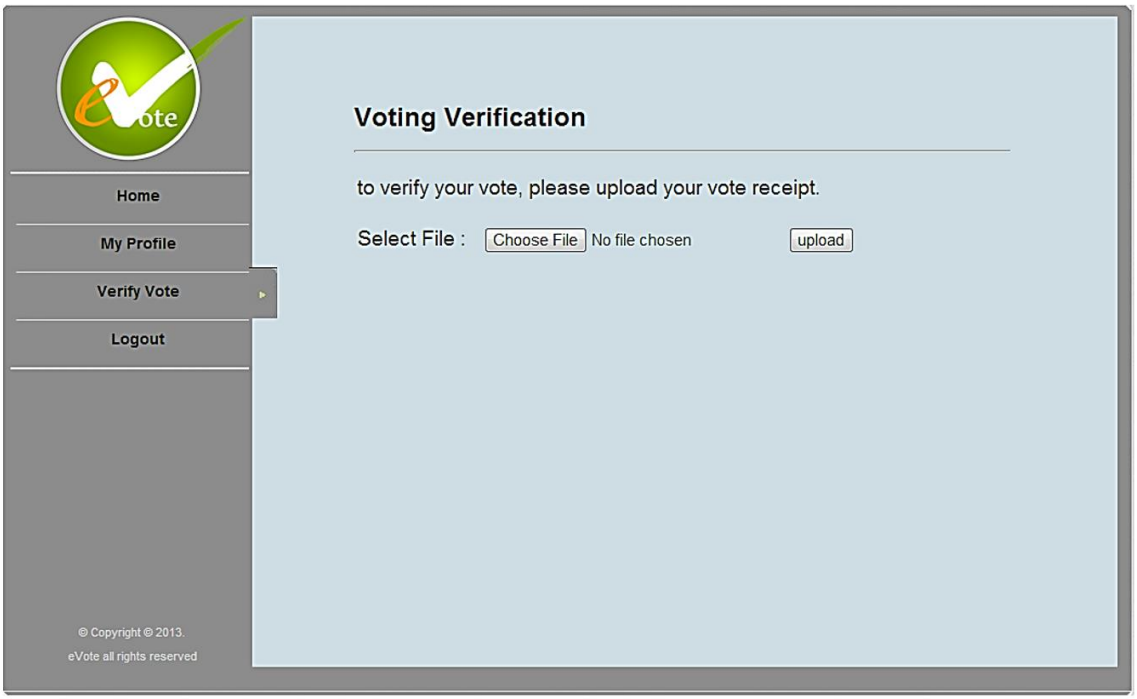

Figure 12. Screenshot of vote verification page for the voters

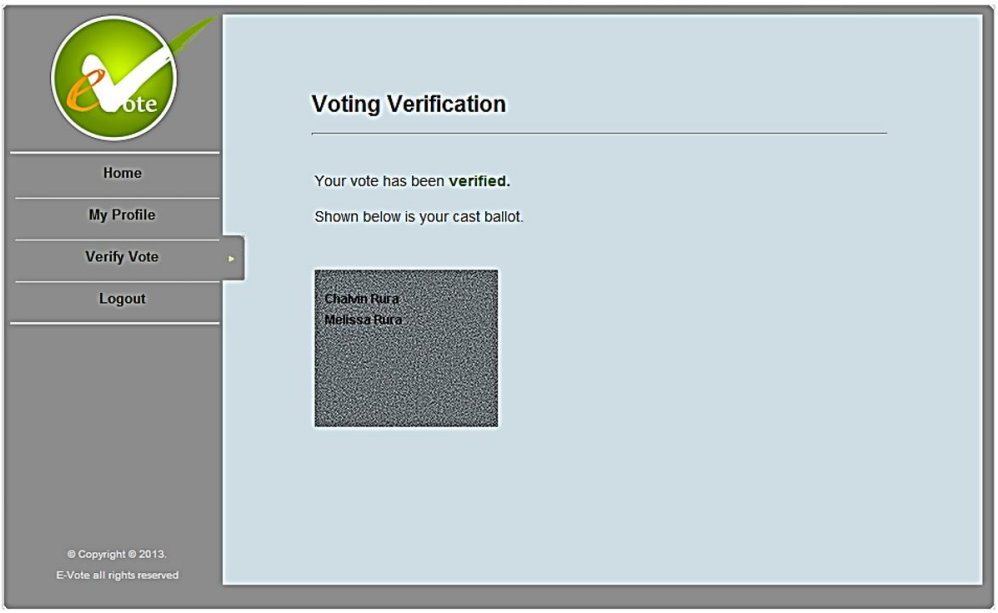

Figure 13. Screenshot of vote verification page upon successful verification

4. SOFTWARE TESTING AND ANALYSIS

This section evaluates the three main properties that need to be ensured by the eVote software besides security namely effectiveness, efficiency and usability. It is done with two distinct types of software testing namely, usability testing and user acceptance testing. Testing the usability of a Moodle-based learning platform is discussed in the paper (Ternauciuc & Vasiu, 2015). In usability testing, the user's experience was examined by assessing Nielsen's five quality components of usability (Nielsen, 2012). On the other hand, user's acceptance was measured by using Davis' Technology Acceptance Model (TAM). Both have been commonly used as a standard for many empirical studies on user experience and acceptance. Here the data is collected through questionnaire before applying Cronbach's alpha test to measure its reliability scale (Cronbach, 1951). For the questionnaire survey, 30 representative individuals from different demographic groups participated. They were recruited based on the consideration of few significant aspects such as gender, the level of education and basic knowledge of information security and usability. The users are also chosen based on the minimum voting age requirement by the Malaysian law. The summary of participant's demographic information is shown in figures 14 and 15. Each of the participants is required to complete a set of voter's tasks assigned to them and also to fill in a questionnaire in not more than thirty minutes. This questionnaire was constructed for the intended users to evaluate the eVote voting software's effectiveness, efficiency and reliability. The test results are discussed in detail in the following sections.

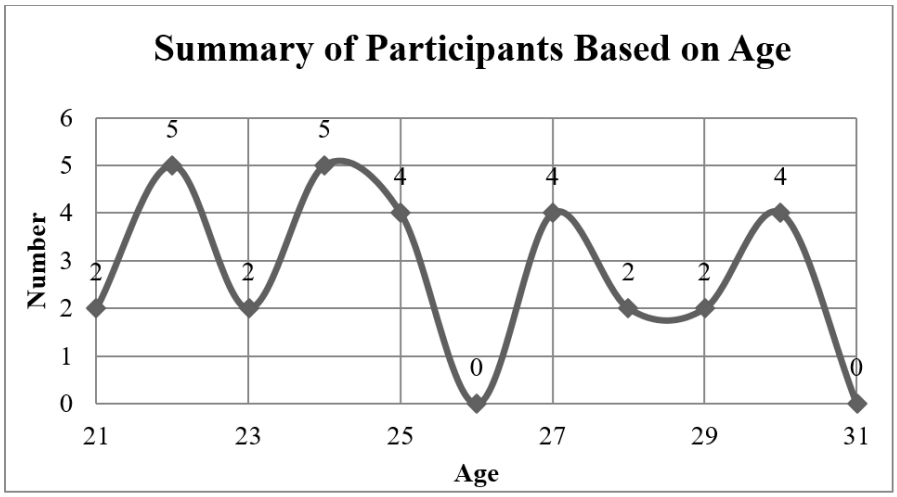

Figure 14. Summary of participants based on their age

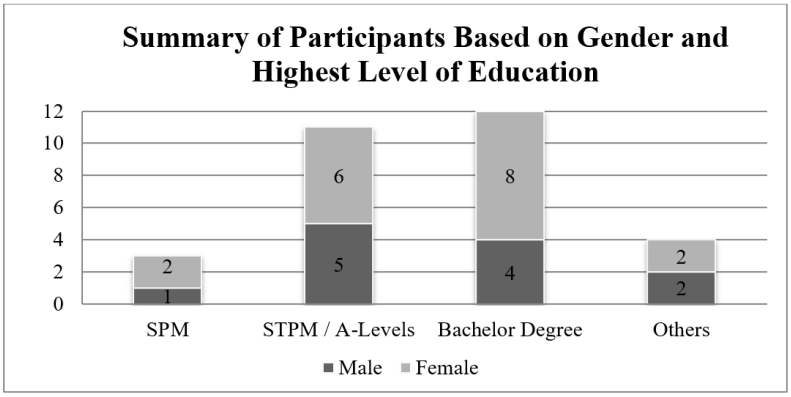

Figure 15**.** Summary of participants based on their gender and highest level of education

Usability Testing

Usability testing measures the concerns of the user about the system. According to Nielsen (2012), there are five quality components that define usability such as - learnability, efficiency, memorability, errors and satisfaction. These five aspects are examined through the following nine out of ten usability heuristics principles for user interface design that Molich and Nielsen had developed (1990).

- Visibility of system status
- Match between the system and the real world
- Consistency and standards
- Aesthetic and minimalist design
- User control and freedom
- Helps user recognise, diagnose and recover from errors
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency to use

From the observation of usability testing carried out by 30 participants with good computer literacy, data collection is done. Illustrated in Table II are its derived results. The information presented in Table II illustrate user's perspective regarding the learnability, efficiency and satisfaction of the system interface. Based on the three sets of the task given to them, they evaluate the navigation process of the eVote software. Users give a positive feedback for their first experience to use the eVote. They can understand the system and navigate through different processes in the three distinct user levels easily. The neat layout with simple, consistent and understandable menu arrangement is one of the factors that supported this.

The other factor that needs be ensured by a system to support user's accessibility is proper error handling. It must be done properly to meet the user requirements. The eVote software provides a number of error-handling mechanisms. JavaScript handles some of them on the client-side, while the others are supported on the server-side. The 30 participants have a satisfactory experience with its error-handling mechanism, and there is room for improvement based on the survey. The last factor that the users evaluated is their effectiveness in using the eVote software for the second time or also referred to as memorability. This is the main reason why they are asked to complete three different set of tasks from each user level of the software. Due to its similar layout design and the straightforward functionalities, it offered, users agree that the software is able to provide efficient election procedures.

User Acceptance Testing

The competency of a software system depends on a lot on user acceptance level. User acceptance testing is conducted to consider the behavioural factors of the users. Based on the data collected we identify the user acceptance level through implementing the TAM by Davis (1989).

In his model Davis claimed that design features are part of the distinct cognitive appraisals of user's attitudes towards using the technology, behavioural intentions to use the technology and the actual usage of the technology as shown in Figure 16 (Davis, 1989).

Table II. Results of the usability testing conducted based on Molich and Nielsen's Usability Heuristics for User Interface Design (Molich & Nielsen, 1990)

Heuristic Principles	Sub-principles	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Visibility of System Status	Ability to understand the interface easily	21	9	0	0	0
	Ability to use the system easily	17	10	3	0	0
Match between the System and the Real World	The function of each icon/button is understandable	8	16	6	0	0
	Ability to follow the order of the system	25	5	0	0	0
Consistency and Standards	Interface layout is arranged in a logical order	22	8	0	0	0
	Poor arrangement of icons/buttons in the interface	0	0	1	6	23
Aesthetic and Minimalist Design	Existence of irrelevant information in the system	0	0	0	10	20
	Experience of user-friendliness from the system design	7	18	2	0	3
User Control and Freedom	Ability to navigate to another page once error is made	8	15	2	5	0
Help User Recognize, Diagnose and Recover from Errors	Ability to handle errors once occurred	0	16	11	3	0
Error Prevention	Errors are occasionally made	0	0	0	19	11
	Ability to handle error once occurred	0	14	12	4	0
Recognition Rather than Recall	Expected functions are available	23	7	0	0	0
	Ability to recognize the function rather than recall	17	13	0	0	0
Flexibility and Efficiency to Use	Capability of inexperienced user to use the system	6	14	10	0	0
	Ability to operate more effectively at the second time	25	5	0	0	0

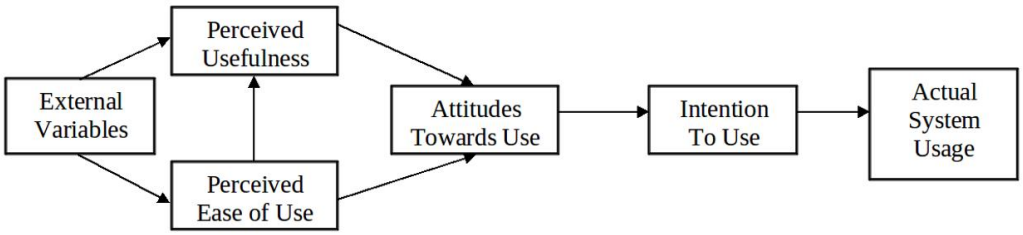

Figure 16. TAM Model (Davis, 1989)

Those design features include Perceived Usefulness (PU) and Perceived Ease of Use (PEU). These are the design features and the responses to the system that we want to measure to determine the user acceptance level. PU (extrinsic motivator) is the degree to which an individual believes that a particular system would enhance his or her job performance. As a measurement tool to obtain PU value, Davis initially proposed six items or criteria. However, there are only four most commonly used criteria, as follows: (1) Using application increases my productivity, (2) Using application increases my performance, (3) Using application enhances my effectiveness on the job and (4) Overall I find the application useful. On the other hand, PEU (intrinsic motivator) is the degree to which a user believes that the use of a particular system would require less effort compared with other systems. Similar to the PU, there are four out of six most commonly used items or criteria that Davis proposed as a measurement tool. The PEU criteria are as follows: (1) Learning to operate the application is easy for me, (2) I find it easy to get the application to do what I want to do, (3) The application is rigid and inflexible to interact with and (4) Overall I find the application easy to use. In this research, the two main aspects of TAM were evaluated based on these eight criteria. The collected data were analysed as follows.

The Cronbach's alpha test was used to determine the internal reliability of PU and PEU of the survey questionnaire. Its results are derived from the equation (1) shown, where n is the number of items, V_i is the variance of item scores, and V_t is the variance of test scores (Cronbach, 1951).

$$\alpha = \frac{n}{n-1}\left(1 - \frac{V_i}{V_t}\right) - (1)$$

These results subsequently determine the reliability of user's acceptance level. Nunnally (1978) and Hair et al. (1998) recommends the value of 0.60 to 0.70 and above as the standard reliability coefficient. However, with the number of items or criteria applied, the coefficient value should be increased accordingly. It should then be calculated with the formula in equation (2), where r_d is the desired reliability, r_e is the reliability of the existing instrument, and k is the number of times the test would have to be lengthened to obtain the desired reliability (Nunnally, 1978).

$$k = \frac{r_d(1-r_e)}{r_e(1-r_d)} - (2)$$

Based on the evaluation conducted, the results of Cronbach's alpha test for eVote on PU and PEU are shown respectively in Table III and IV. Both portrays good results of 0.89 for PU's measurement and 0.88 for PEU's measurement. These values are higher than the benchmark values of 0.6 to 0.7 that is set as standard, which means the results of the questionnaire conducted is reliable. Specifically, the user's acceptance of the system can be concluded as very good. From the comments gathered during the survey, most of the participants prefer to use the remote E2E voting system, compared to the polling booth provided in a traditional voting system to cast their votes. This is mostly due to the convenience

and practicality it offered. According to them, the implementation of vote receipt is more reliable and offers more assurance to them, compared to the implementation of indelible ink commonly used in the traditional voting system. There are many ways counterfeit votes can be cast using indelible ink.

On the other hand, the implementation of vote receipt only requires the involvement of system administrators and polling officers who are assumed to be trustworthy. Besides that, most of the participants are also assured that their votes have been counted as cast and are kept securely by the eVote voting system. From the survey conducted, we also concluded that vote receipt in visual cryptography image format is more preferable compared to ciphertext format. It is more practical and convenient to be used by the voters.

Table III. Results of the user acceptance test (PU's measurement) conducted based on Davis TAM (Davis, 1989)

PU Items	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree	Rate Variance
Using application increases my productivity	5	9	10	4	2	**11.5**
Using application increases my performance	7	11	8	3	1	**16**
Using application enhances my effectiveness on the job	9	17	2	2	0	**49.5**
Overall I find the application useful	10	12	3	3	2	**21.5**
Total	**31**	**49**	**23**	**12**	**5**	**98.5**
Mean	**24**					
SD	**17.18**					
SD of TAM Items	**295**					
Cronbach's Alpha Value	**0.89**					

5. CONCLUSION AND LIMITATIONS

The development of the eVote system addresses the common problems that arose in the traditional voting procedure. The main contribution of this work lies in the simplicity and user-friendliness the eVote software offers without compromising system security, efficiency and performance. The usability and user acceptance testing done has shown that the developed system is favoured by the

users. The technology behind the voting system is as follows. Password hashed-based scheme was applied to secure user's password in registration and authentication stage. The plaintext of the ballot was encrypted using visual cryptography to two shares of ciphertext. One share was given to the voter as their vote receipt, and the other share was stored in the database. Visual cryptography decryption algorithm was executed to decrypt these shares. It also provides improved vote receipt mechanism. The vote receipt in our system is shown as visual cryptography image format. The voter's receipt is segmented into two parts. The voters only possess a part of that receipt, and the other part is kept secure in the database. The only individual who has access to their votes is the voters themselves, and it can only be done through the vote verification feature. Threshold decryption scheme was implemented in the ballot decryption process of the tallying stage to ensure that only authorised personnel can have access to the vote tallying process. F5 image steganography provided an enhanced security technique of data encoding with the digital image file as the cover file. Image steganography is used to ensure voters' receipts are only accessible to the voter themselves. The system works well assuming that the server used to store the information and the authorities involved are trustworthy.

Table IV. Results of the user acceptance test (PEU's measurement) conducted based on Davis TAM (Davis, 1989)

PEU Items	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree	Rate Variance
Learning to operate the application is easy for me	9	16	3	1	1	**42**
I find it easy to get the application to do what I want to do	6	18	3	2	1	**48.5**
The application is flexible to interact with	6	7	14	3	0	**27.5**
Overall I find the application easy to use	8	17	3	1	1	**46**
Total	**29**	**58**	**23**	**7**	**3**	**164**
Mean	**24**					
SD	**21.86**					
SD of TAM Items	**478**					
Cronbach's Alpha Value	**0.88**					

On the limitation side, there are security attacks possible on the online voting system. If a group of polling officers work in unethical ways, the system can be compromised. But one polling officer alone cannot compromise the system because to access the voting tally list, the polling officers must perform

'decryption' process by merging their unique secret keys. Only after each of the polling officers has submitted their unique secret keys, the result tally list is accessible to the system administrators and the polling officers for monitoring. The distributed denial of service (DDoS) attacks could overload the servers on the election day, the hackers could eavesdrop the network traffic and could potentially impersonate normal users through spoofing attacks to cast false votes. The web application could be attacked through shell-injection vulnerability or other web technology flaws. The attackers could get through by attacking the payloads and through attacking the network infrastructure. There are other attacks possible on specific technologies used like F5 image steganography and visual cryptography that we have not discussed. But there are various countermeasures that can be implemented to secure the network and network resources (Haynes, 2014), including an additional layer of biometric authentication. These discussions can be long and are beyond the scope of this paper.

Future direction of research can be focused on implementing a more secure online voting system that uses multiple levels of security with better technologies and that which would address all the security challenges of the network upon which it is implemented. Though a fool-proof and perfect online voting is impossible in theory, the overall security can be hardened in as many ways as possible.

REFERENCES

Adida, B. (2008). Helios: Web-based Open-Audit Voting. In *Proceedings of the 17th Conference on Security Symposium*, USENIX Association, Berkeley, USA, pp. 335-348.

Aditya, R (2005). Secure Electronic Voting with Flexible Ballot Structure. *PhD Thesis*, Faculty of Information Technology, Queensland University of Technology, Australia.

Ambler, S.W. & Sadalage, P.J. (2006). *Refactoring Databases: Evolutionary Database Design*, Addison-Wesley Professional.

Azougaghe, A., Hedabou, M. & Belkasmi, M. (2015). An electronic voting system based on homomorphic encryption and prime numbers. In *Proceedings of the 11th International Conference on Information Assurance and Security (IAS)*, Marrakech, Morocco, pp. 140-145.

Bateman, P. (2008). Image Steganography and Steganalysis, *Master's Thesis*, Faculty of Engineering and Physical Sciences, University of Surrey, UK

Benaloh, J. (2006). Simple Verifiable Elections. *In Proceedings of the USENIX/Accurate Electronic voting Technology Workshop 2006 on Electronic Voting Technology Workshop*, USENIX Association, Berkeley, USA, pp.5-5.

Beroggi, G. E. G. (2014). Internet Voting: An Empirical Evaluation, *Computer*, 47(4), 44-50.

Burton, C. Culnane, C. & Schneider, S. (2016). vVote: Verifiable Electronic Voting in Practice, *IEEE Security & Privacy*, 14 (4), 64-73.

Carback, R., Chaum, D., Clark, J., Conway, J., Essex, A., Herrnson, P.S., Mayberry, T., Popoveniuc, S., Rivest, R.L., Shen, E., Sherman, A.T. & Vora, P.L. (2010). Scantegrity II Municipal Election at Takoma Park: The First E2E Binding Governmental Election with Ballot Privacy. In *Proceedings of the 19th USENIX Conference on Security*, USENIX Association, Berkeley, USA, pp. 19.

Cetinkaya, O. (2008). Analysis of Security Requirements for Cryptographic Voting Protocols, In *Proceedings of Third International Conference on Availability, Reliability and Security 2008*, IEEE Educational Activities Department, Piscataway, USA, pp. 1451-1456

Chaum, D. (2004). Secret-Ballot Receipts: True Voter-Verifiable Elections, *IEEE Security and Privacy*, 2(1), pp. 38-47.

Chaum, D., Carback, R., Clark, J., Essex, A., Popoveniuc, S., Rivest, R.L., Ryan, P.Y.A., Shen, E. & Sherman, A.T. (2008). Scantegrity II: End-To-End Verifiability for Optical Scan Election Systems Using Invisible Ink Confirmation Codes. In *Proceedings of the Conference on Electronic Voting Technology*, USENIX Association, Berkeley, USA.

Cronbach, L. J. (1951). Coefficient Alpha and the Internal Structure of Tests. *Psychometrika*, 16(3), pp. 297-334.

Davis, F. D. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13(3), pp. 319-340.

Fard, A. M., Akbarzadeh-T, M. R., & Varasteh-A. F. (2006). A New Genetic Algorithm Approach for Secure JPEG Steganography. In *Proceedings of IEEE International Conference on Engineering of Intelligent Systems*, Islamabad, pp. 1-6.

Fridrich, J., Goljan, M. & Hogea, D. (2002). Steganalysis of JPEG Images: Breaking the F5 Algorithm. In *Proceedings of the 5th International Workshop, IH 2002 Noordwijkerhout*, The Netherlands, pp. 310-323.

Fujioka, A., Okamoto, T. & Ohta, K. (1992). A Practical Secret Voting Scheme for Large Scale Elections. In *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques: Advances in Cryptology*, Springer-Verlag, London, UK, pp.244-251.

Gritzalis, D. A. (2002). Principles and Requirements for a Secure E-Voting System. *Computers & Security*, 21(6), pp. 539-556.

Hair, J. F., Anderson, R. E., Tatham, R. L., & Black, W. C. (1998). *Multivariate analysis*. Englewood: Prentice Hall International.

Haynes, P. (2014). Online Voting: Rewards and Risks, Atlantic Council, Intel Security, Washington DC. Retrieved from http://www.mcafee.com/us/resources/reports/rp-online-voting-rewards-risks.pdf

Hubbers, E., Jacobs, B. & Pieters, W. (2005). RIES — Internet Voting in Action. In *Proceedings of the 29th Annual International Computer Software and Applications Conference*, IEEE Computer Society, Washington DC., USA, pp. 417-424.

Kofler, R., Krimmer, R. & Prosser, A. (2003). Electronic Voting: Algorithmic and Implementation Issues. In *System Sciences Proceedings of the 36th Annual Hawaii International Conference*, IEEE Computer Society, Washington DC., USA.

Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing* Systems, 1990, ACM, New York, USA, pp. 249-256.

Morkel, T., Eloff, J. H., & Olivier, M. S. (2005). An overview of image steganography, In Proceedings *of the Fifth Annual Information Security South Africa Conference in Sandton, South Africa*, pp. 1-11.

Naor, M. & Shamir, A. (1994). Visual Cryptography. *Workshop on the Theory and Application of Cryptographic Techniques*, In *Proceedings of Lecture Notes in Computer Science*, Springer-Verlag, pp. 112.

Nielsen, J. (2012). Usability 101: Introduction to Usability. Retrieved from: http://www.nngroup.com/articles/usability-101-introduction-to-usability/

Nunnally, J. (1978). *Psychometric methods.* McGraw-Hill, New York, NY.

Petcu, D. & Stoichescu, D. A. (2015). A hybrid mobile biometric-based e-voting system. In Proceedings of 9th International Symposium on Advanced Topics in Electrical Engineering (ATEE), Bucharest, pp. 37-42.

Provos, N & Honeyman, P (2003). Hide and seek: An Introduction to Steganography. *IEEE Security & Privacy*, 1(3), pp. 32- 44.

Rura, L., Issac, B. & Haldar, M. K. (2011). Online Voting Verification with Cryptography and Steganography Approaches. In *Proceedings of IEEE International Conference on Computer Science and Network Technology 2011 (ICCSNT 2011)*, pp.125-129.

Rura, L., Issac, B. & Haldar, M. K. (2011). Analysis of Image Steganography Techniques in Secure Online Voting. In *Proceedings of IEEE International Conference on Computer Science and Network Technology 2011 (ICCSNT 2011)*, pp.120-124.

Ryan, P.Y.A., Bismark, D., Heater, J., Schneider, S. & Zhe Xia (2009). Prêt à Voter: a Voter-Verifiable Voting System. *IEEE Transactions on Information Forensic and Security*, 4(4), pp. 662-673.

Ryan, P. Y. A., Schneider, S. & Teague, V. (2015). End-to-End Verifiability in Voting Systems, from Theory to Practice, *IEEE Security & Privacy*, 13 (3), 59-62.

Shamir, A. (1979), How to Share a Secret. *Communications of the ACM*, Vol. 22, No. 11, pp. 612-613.

Ternauciuc A. & Vasiu, R. (2015). Testing usability in Moodle: When and How to do it. In *Proceedings of the IEEE 13th International Symposium on Intelligent Systems and Informatics (SISY)*, Subotica, pp. 263-268.

Wagner, D. & Goldberg, I. (2000), Proofs of Security for the Unix Password Hashing Algorithm. In *Proceedings of Lecture Notes in Computer Science*, Springer-Verlag, London, UK, pp. 560-572.

Wang, H., & Wang, S. (2004), Cyber Warfare: Steganography vs. Steganalysis, *Communications of the ACM*, 47 (10), pp. 76-82.

Westfeld, A. (2001). F5- A Steganographic Algorithm. In *Proceedings of the 4th International Workshop on Information Hiding (IHW '01)*, Springer-Verlag, London, UK, pp. 289-302.

Biographies:

Lauretha Rura received her Master of Science (by research) from Swinburne University of Technology (Sarawak Campus), Malaysia. Her main research topic is the enhancement of E-voting system, E2E verifiable voting system in particular.

Biju Issac is working in Teesside University as an academic staff. He earned a PhD in Networking and Mobile communications, along with MCA (Master of Computer applications) and BE (Electronics and Communications engineering). Dr Issac is research active and has authored more than 70 refereed conference papers, journal papers and book chapters. He is in the technical programme committee of many peer-reviewed international conferences and journals.

Manas Kumar Haldar has been with the Swinburne University of Technology, Sarawak Campus since 2006. He has obtained his PhD as Charles Hestermann Merz scholar of Trinity College, Cambridge, UK. He worked on high frequency power generation by electron wave interactions. He also worked on surface acoustic waves at the University of Oxford, UK. He has over 30 years of teaching and research experience. He is a reviewer of many conferences and journals, such as the Asia Pacific Microwave Conference and the Journal of Light wave Technology.