CIT. Journal of Computing and Information Technology, Vol. 25, No. 1, March 2017, 47–61
doi: 10.20532/cit.2017.1003224

Online Voting System Based on Image Steganography and Visual Cryptography

Lauretha Rura[1], Biju Issac[2] and Manas Kumar Haldar[1]

[1]Swinburne University of Technology (Sarawak Campus), Malaysia
[2]Teesside University, Middlesbrough, United Kingdom

This paper discusses the implementation of an online voting system based on image steganography and visual cryptography. The system was implemented in Java EE on a web-based interface, with MySQL database server and Glassfish application server as the backend. After considering the requirements of an online voting system, current technologies on electronic voting schemes in published literature were examined. Next, the cryptographic and steganography techniques best suited for the requirements of the voting system were chosen, and the software was implemented. We have incorporated in our system techniques like the password hashed based scheme, visual cryptography, F5 image steganography and threshold decryption cryptosystem. The analysis, design and implementation phase of the software development of the voting system is discussed in detail. We have also used a questionnaire survey and did the user acceptance testing of the system.

ACM CCS (2012) Classification: Security and privacy → Security services → Authentication → Graphical / visual passwords

Security and privacy → Security services → Access control

Keywords: electronic voting system; image steganography; visual cryptography; user acceptance testing

1. Introduction

Many people nowadays prefer a faster and more secure way to vote. A traditional election procedure does not meet voters' demands anymore. This might lead to a low turnout in an election. With the rapid growth of computer technology, many researchers are proposing secure, reliable, and convenient electronic voting systems as a substitute to the traditional voting procedure. Compared to the traditional voting system, the new system could offer shorter time in finishing the main operations in the election progress, such as vote casting, authentication, registration and vote tallying processes. In a traditional voting procedure, all these processes would be divided into few stages at few different stations. Implementation of the electronic voting system could help the election officials who administer the election and minimise the cost of the election itself. If an electronic voting system is designed properly, it can also provide a more secure system than the paper-based election by providing precise data communication and preventing threats and attacks by intruders.

In recent years, researchers are focusing more on developing new technologies that can prevent coercion, provide receipts to ensure voter-verifiability and offer universal-verifiability through the implementation of the bulletin board. These three aspects are the vital components of a reliable election procedure. The voting systems with such characteristics are categorised as end-to-end verifiable voting systems (E2E). Many E2E systems have been proposed and are widely used nowadays [1] – [7]. In principle, an E2E voting system offers assurance to the voters by distributing a receipt of their vote after they have cast their votes, which can be used for verification purpose from the overall tabulation of the collected votes. This receipt could not be used as a proof of vote buying or coercion, but all of those encrypted receipts will be posted publicly in a read-only Bulletin Board after each voter finishes the voting process. Thus, the E2E system could still protect the voter's privacy and ensure election's integrity. The proposed E2E voting system in this work is based on the earlier design concepts, but differs from the previous work because of the im-

plementation of two distinct schemes. They are cryptography and steganography. Both components of information security are combined in a layer of data protection. In electronic voting, cryptography is a commonly used technique as it is a good defence against threats. However, steganography has not been used commonly as an additional layer of security in an online voting system. The combination of these two schemes is the novel approach proposed in this work, mainly implemented to secure the communication between the user and the server. It is expected to produce an improved technique, which could meet the voter's demand and perform with a less performance cost in a secure manner.

To ensure the integrity of an election, many schemes have been implemented and proposed. In these schemes, cryptography is used to protect the data transmitted between the voter and the server, to ensure that it would not be leaked to a third party. The cryptography techniques are also applied in each process in the system to ensure the authenticity of the voter, the originality of the ballots cast and collected votes, the reliability of the tallied votes and the privacy throughout the election. There are many cryptography methods that can be applied, such as blind signature scheme, homomorphic encryption, oblivious signature scheme, bit commitment scheme, Schnorr identification scheme, mixed-net schemes, digital signature scheme, secure multi-party computation, cryptographic hash-function, etc. However, in this work, only a few selected schemes are used. These are applied in different voting stages to preserve the main characteristics of an electronic voting system. The selected schemes are password hashed-based scheme, visual cryptography [8] as adopted from secret-ballot receipts proposed by Chaum [9] and threshold decryption cryptosystem.

Steganography, which is a branch of information security technique and has not been commonly used in E2E voting systems, is also included in the software architecture design. It is the science of hiding information in communications, where no one other than the sender and receiver would know the existence of hidden information [10]. In 2007, Hong and Hong stated that steganography pays less attention to intentional attacks since it focuses more on data insertion capability [11]. However, as information technology evolved and more threats arose, it became necessary to develop more secure steganography algorithms. The advantage of steganography over cryptography is its ability to offer a more advanced way of hiding a secret. Therefore, steganography is used to secure the data communication in this research. This scheme provides secret communication accessible by encoding a secret message to various types of cover data such as text, images, audio, video file format. Each covered data has multiple methods to hide the secret message. Unlike cryptography, the output data of steganography (stego-object) would still look the same as its input data. Thus, it would be difficult to identify and interpret the hidden secret in the stego-object. For electronic voting system implementation, both image and text steganography are appropriate candidates. They have more redundancy, which allows larger size data to be encoded into the cover file. Other than that, they are also unlikely to raise any suspicion because they are the most common data transmitted between the voter and the server. However, image steganography offers a better encoding technique to be used as it can hide the secret message by securely transferring a hidden secret in a digital image file [12] – [13]. Hence, the implementation of image steganography is used in this work.

There are five different stages in the system design architecture, namely the registration stage, authentication stage, voting stage, tallying stage, verification stage. The secret-ballot receipts theorem introduced by David Chaum [9] is mainly a combination of cut and choose scheme together with a cryptography technique called visual cryptography. This scheme is applied and modified in this work. Thus, visual cryptography will be implemented in the voting stage and verification stage as part of secret-ballot receipt implementation. Right after a vote is done, steganography will be used throughout the system processes for data communication purpose. In the tallying stage, the threshold decryption cryptosystem will be implemented. The combined method is believed to be sufficient to provide a secure, reliable and convenient voting system. Since the proposed tool is an electronic voting system, it is necessary to assume that the voter would complete the voting process secretly. On a basic level, the E2E voting framework offers confidence to

the voters in their voting process. This is implemented by circulating the vote receipt of encoded voting done for verification. To bolster this confirmation procedure, E2E frameworks executed some form of secure announcement board where each of the encoded votes would be posted once the voters finished the voting procedure. To check their cast votes, they have to match the encoded value on their receipt against the values published on the announcement board. However, the vote receipt cannot be utilised as a proof of vote purchasing or vote coercion since it is encoded. Accordingly, the E2E voting framework would secure the voter's protection and backs incoercibility that supports the trustworthiness and fairness of the election result.

To achieve the aims and objectives of this work, the Software Development Life Cycle (SDLC) was used to implement the eVote system, a secure end-to-end verifiable electronic voting system. The primary SDLC method is known as the waterfall method. An improved version of this method, the iterative waterfall model has been used for the eVote system development as it allows system developers and designers to correct any mistakes made at any of the stages during the system development. The iterative waterfall model is divided into five distinct phases, namely requirement analysis, design, implementation, testing and maintenance. This paper is organised as follows. Section 2 is the related works, Section 3 is the software requirement analysis of the voting system, Section 4 is the software design of the voting system, Section 5 is the implementation of the voting system, Section 6 is the user acceptance testing of the voting system, and Section 7 is the conclusion.

2. The Related Works

To better understand the previous works, we studied four E2E voting systems, namely Helios open-source web-based voting system, Scantegrity II optical voting system, Prêt à Voter and RIES voting system to better understand the features of the online voting systems [1], [14] – [17].

Helios is an open-source web-based voting system that offers verifiable online elections for anyone. It was designed to ensure a clean election setting with ballot secrecy and election integrity. Scantegrity II is a practical enhancement for optical scan voting systems, which achieve increased election integrity through a novel use of confirmation codes printed on ballots in invisible inks. The development of Prêt a Voter was initially motivated by Chaum's work of implementing visual cryptography approach proposed by Naor and Shamir [18] in an election in 2004 to offer vote verifiability. It implements the same concept as Chaum's secret-ballot receipt scheme to provide more accurate and faster tallying process, to cut unnecessary election cost and to increase voter participation. Prêt à Voter offers assurance from its election auditability. RIES is a voting system that implements multiple types of technologies. 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 result. They were used in actual governmental and organisational elections, two of which are used for elections in educational institutions. All E2E voting systems were designed to fulfil two main objectives of E2E voting systems, to provide individual-verifiability (also known as voter-verifiability) and universal-verifiability. The four voting systems reviewed are equipped with both features. Ironically, the implementation of these features caused the initiation of some known attacks like randomization attack.

We have looked at some other works as well. Santos, de Queiroz, Saraiva and Junior [19] analysed various processes of electronic voting and counting of votes, to better understand the secrecy of the vote. Chondros, Zhang, Zacharias, Diamantopoulos, Maneas, Patsonakis, Delis, Kiayias and Roussopoulos [20] present the design, implementation, security analysis, and evaluation of D-DEMOS e-voting system. It is a distributed, privacy-preserving and end-to-end verifiable voting system. Sultan, Barbhuiya and Sarma [21] discussed an online voting scheme by using biometric and password based security that makes use of fuzzy extractor to provide biometric based authentication, while the secret password is used to provide password-based protection of the voter, along with pairing-based cryptography to provide the needed security. Tornos, Salazar and Piles [22] describe the implementation of a secure

eVoting system, based on ring signatures providing multiple features such as linkability or anonymity. Yi, Wang, and Ma [23] proposed a security sequencing protocol based on homomorphic encryption as secure multipart computation is becoming more and more popular in anonymous voting and online auction. Naidu and Kharat [24] discussed a secure authentication based on biometric features that use visual cryptography to provide confidentiality to the biometric database. It also uses the hash of a number to embed into image shares using cryptography and steganography.

3. Software Requirement Analysis of the Voting System

A system has certain requirements that need to be satisfied for it to be able to function properly. The eVote as a remote E2E voting system not only offers secure and reliable voting system, but it also provides a flexible platform for the election officials to set up and maintain based on their needs. The system users are divided into three distinct types (levels) as follows, based on the user access rights.

3.1. Voters

In every electoral system, may it be traditional or electronic, the voters are one of its primary users. They are the main motive behind electronic voting system's development. The voters expect and request a secure system that is time efficient and easy to use. With the evolution of information technology, it is possible to develop such a system.

3.2. Polling Officers

Adopted from the traditional voting system's procedure, polling officers are needed in kiosk electronic voting system or in any voting system that provides voting at polling booths (stations). The remote electronic voting system normally does not require assistance from the polling officers. The electronic voting systems can add another layer of security where polling officers are employed for protection from possible threats.

3.3. System Administrators

Every system is required to have system administrators to manage and maintain the system. This is true in electronic voting systems as well. In eVote, the administrators can set up an election, add eligible voters and officers, edit the details of registered voters and registered officers, assign different voters to their respective polling officers, etc. Even with the highest level of access, it is not possible for the eVote's system administrators to cheat or to execute some known attacks on the electronic voting system.

The eVote is an improved version of the existing end-to-end verifiable voting system. The various features supported by eVote are subject to the user type, with the system administrators as the highest user level. The functional requirements are a set of tasks or functions the system is required to perform like in any voting system. It summarises the intended behaviour of the system. The eVote's non-functional requirements consist of its security, usability, performance and reliability. The functional requirements of the eVote system include system authorization, system authentication, applied technologies, system documentation, its data type and data handling. The non-functional requirements are a set of tasks or functions that shall be performed by the system. It sums up the whole operations of the system.

Some of the user requirements gathered in the requirements phase of the online voting software development are as follows: mobility and convenience, completeness (where all valid votes are counted correctly), eligibility and un -reusability (where no voter can vote twice), privacy, soundness (where a dishonest voter cannot disrupt the voting), verifiability (where no one can falsify the result of voting) and robustness (where the result reflects all submitted and well-formed ballots correctly), incoercibility (where only the voters can acquire any information regarding their secret ballots) and receipt-freeness (where each voter can neither obtain nor be able to construct a receipt) and fairness. We will show how these were addressed at the end of this paper. The screenshot of polling officer's homepage is shown in Figure 1.

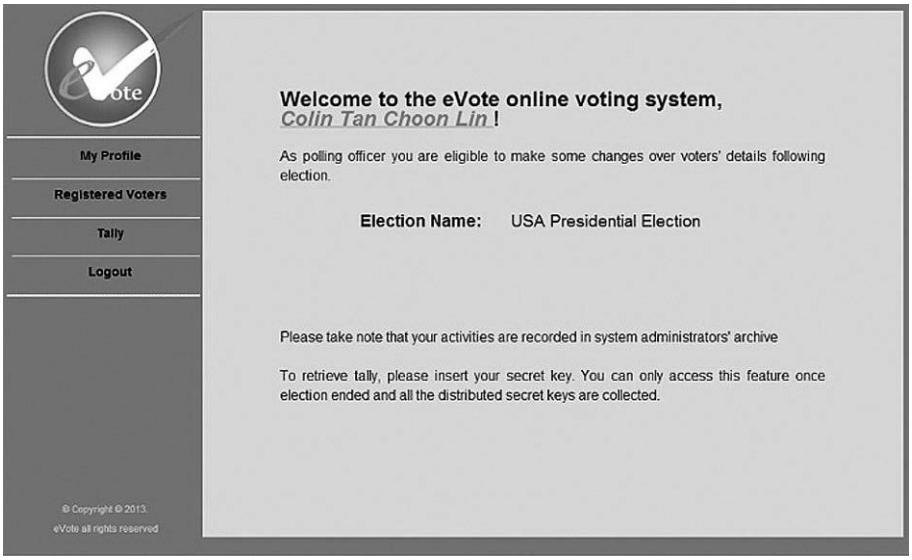

Figure 1. Screenshot of polling officer's homepage.

4. Software Design of the Voting System

The software design of the eVote system is divided into few parts like system integration and use case diagrams. System integration elaborates the overall integration between the components of the eVote voting system. It is described in more detail in the graphical notation of the models. They are discussed as follows.

One of the requirements that eVote must meet is mobility requirement. To fulfil this requirement, the eVote voting system must be built as a web application. Hence, the votes can be cast from anywhere the users prefer, provided stable Internet connection is available.

Due to its low platform dependency as well as other characteristics such as security, robustness and scalability, Java EE 6 has been chosen as the main platform of the eVote's system architecture. Java EE infrastructure is a set of specifications implemented by different containers [25]. The containers themselves are Java EE runtime environments that support a set of Application Programming Interface (API) and provide various services to their hosted components. The web container offers services for managing and executing web components. In this work, they are the modules of the eVote system. Each module carries out a task that can include or extend other tasks the system performs. In the use case diagram, the modules are represented as actions. The use case diagram could be used to conveniently document the system activities. It shows the roles of the eVote voting system and how the system implements those roles [26], as shown in Figure 2.

The web containers are also required to instantiate, initialize and invoke servlets and support HTTP and HTTPS protocols [25]. EJB container, on the other hand, manages and executes enterprise beans that contain the eVote's business logic. It creates and manages the instances of EJBs and JPAs and provides a number of services for eVote, such as distributed transaction between network hosts, security, database access, naming and directory service [25]. Its integration with the client machine, Java application server and the database server are shown in Figure 3.

5. Implementation of the Voting System

The eVote system implementation is done as two versions based on the different Java platforms used for the implementation, and the two versions are eVote v1.0 and eVote v2.0. They are described as follows. The eVote v1.0 is not equipped with registration, authentication, tallying and publishing stages. It was mainly developed to examine the reliability of the implemented primary security approaches used, i.e. image steganography and visual cryptography.

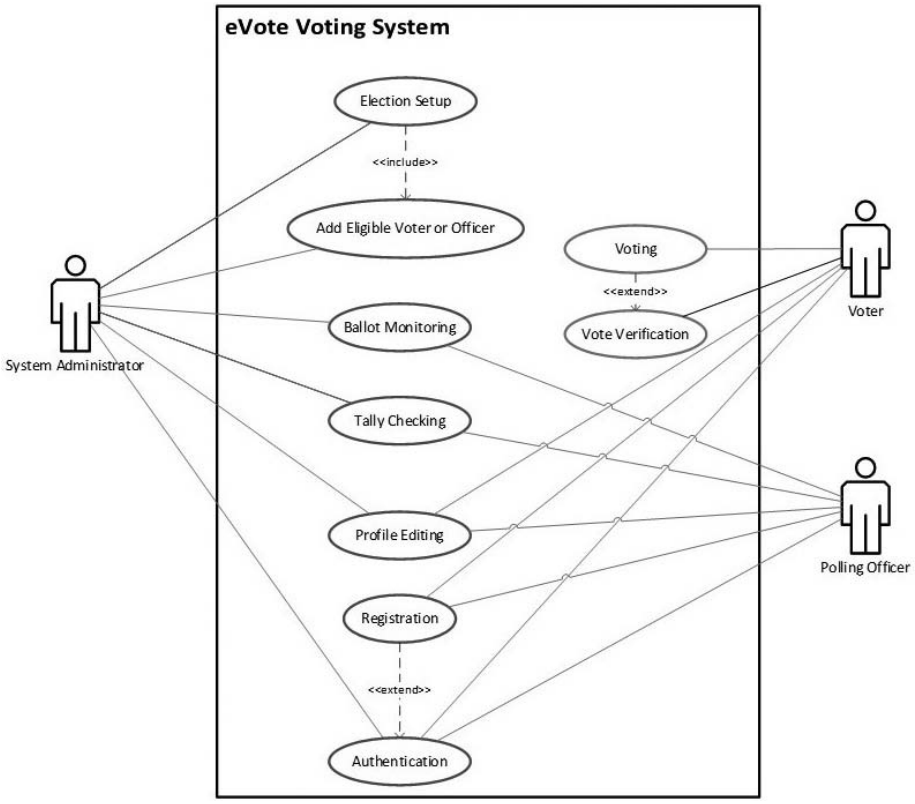

Figure 2. eVote's business processes in use case diagram.

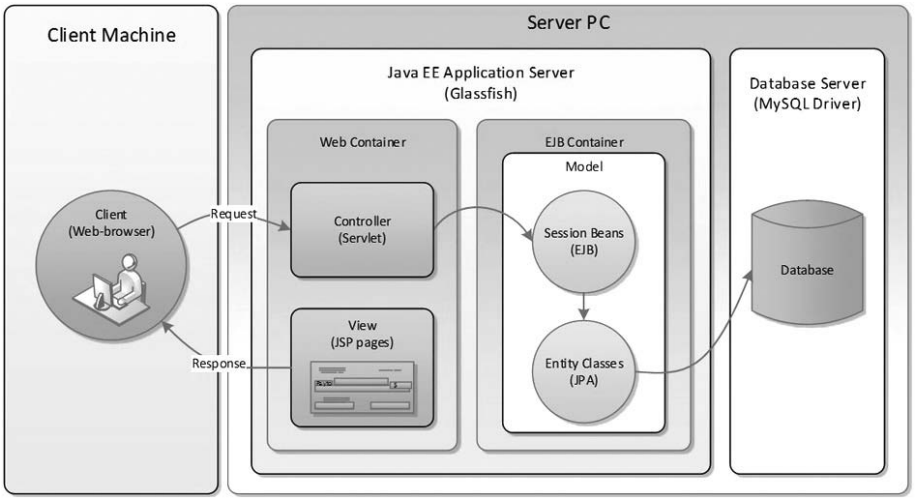

Figure 3. Integrated eVote system.

The eVote v1.0 was developed using Java Standard Edition (SE) platform with no connection to the Database Server. However, eVote v2.0 is a complete voting system. In version 1, a few JFrame GUIs were implemented to perform the main components of the eVote's business process at a single user level to accommodate quality assurance of the selected security approaches. The voting stage is briefly described as follows. To secure the vote transmission between the client and server and to assure the user of the integrity of the votes, both F5 image steganography algorithm and visual cryptography scheme are implemented in this stage. The

system must create a compatible source image containing the ballot for visual cryptography encryption by passing a secret message derived from the decrypted stego-image. Because of the visual cryptography encryption, half of the shares need to be obtained by the voters [27] – [28].

They are prompted to complete this task by saving their vote receipt into the folder they specify. The vote verification is ensured as follows. 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. Thus, 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 formed 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 voters' ballot. The voters in such a system can verify their votes by submitting the vote receipt sent to their email accounts. The verification feature is supported by the 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.

The result of this decryption mechanism is the voters' ballot. In this way, most of remote electronic voting system's requirements, such as incoercibility, receipt-freeness, universal-verifiability, etc. are ensured as neither the polling officers nor the system administrator has access to identifying the collected ballots. Only the voters themselves have access to them. The eVote v2.0 is an enhanced version of eVote v1.0. The eVote v2.0 is developed based on the preliminary studies done in this work. It is developed using Java EE 6 framework. The system is equipped with three types of user levels, i.e. voters, polling officers and system administrators. The eVote v2.0 offers full implementation of the system with many features. The processes of vote verification in both versions of eVote are identical. However, the voting process of eVote v.10 and v2.0 are different due to the implementation of the database. Further explanation of the four stages of the eVote system is given as follows. They are the registration stage, authentication stage, voting stage and tallying stage.

5.1. Registration Stage

This stage is also known as the preparation stage. In this stage, all constraints for the election are prepared. The registration stage is completed by the voters and polling officers. Before this stage, system administrators must prepare the election setup by inserting the details of the election and the candidate's details for each category in the election. Besides, the system administrators also need to add eligible voters' and polling officers' records in the database. This record includes their usernames, Identification Card (IC) numbers and their valid email addresses. Upon successful attempts, eligible users will receive an email from the system administrator notifying their eligibility to register into the eVote system. Only eligible users can carry out the voter's registration. By accessing the link provided in the email, eligible voters and officers can now register themselves in the system. To register themselves in the system, users are required to provide their details and submit them to the system and ensure the accuracy of the details given by the users. These are then matched with users' details saved by the administrators in the database. As another layer of security, users' passwords will be cryptographically secured by applying password hashed-based scheme. By the implementation of this cryptography technique, only the hashed password together with its salt value is saved into the database. Its implementation will be explained further in the next stage of the eVote system, which is authentication stage. Figures 4 and 5 show the registration process flow of the voters in two distinct sequence diagrams. There is a slight difference between voter and officer registration processes. 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 the successful registration process, users will be directed to their respective homepage by the system.

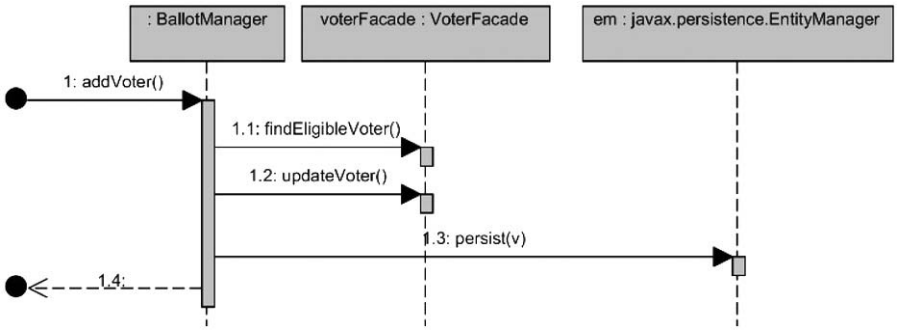

Figure 4. Sequence diagram of voter registration process.

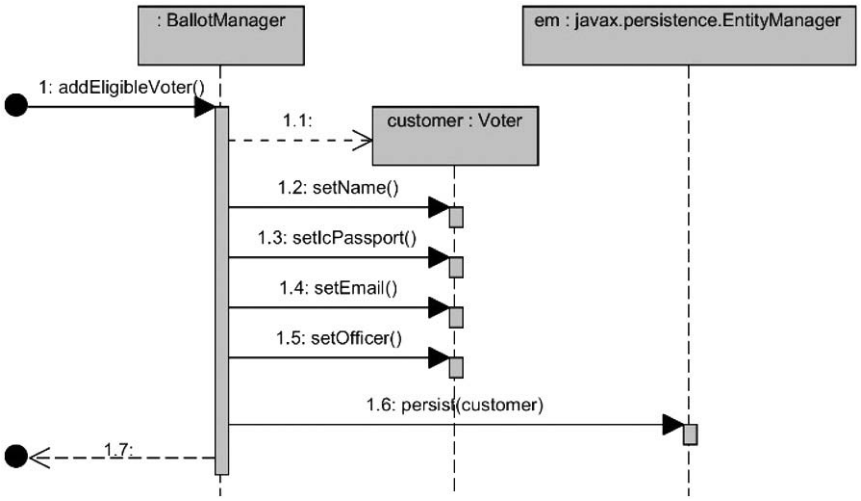

Figure 5. Sequence diagram of addEligibleVoter() function.

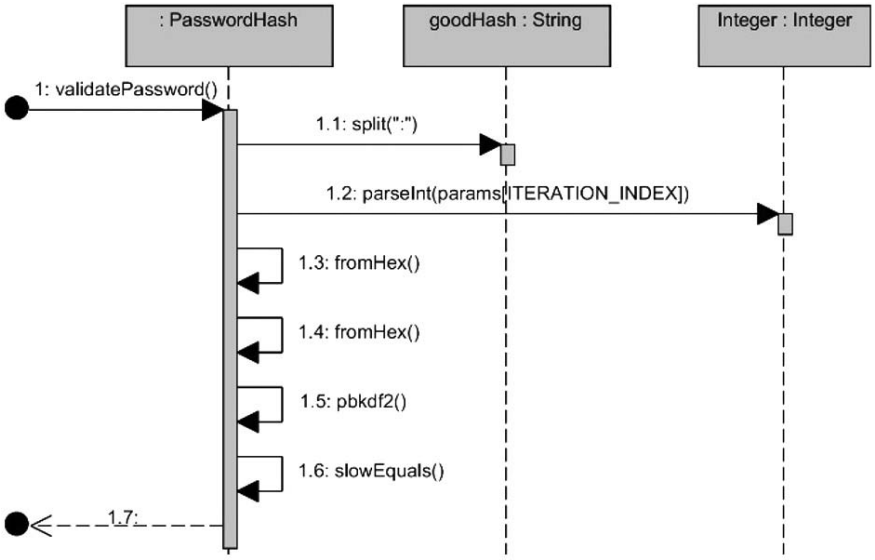

Figure 6. Sequence diagram of user authentication process.

5.2. Authentication Stage

In a remote electronic voting system, the implementation of this stage is mandatory. The objective of this stage is to ensure voters' identity. Registered voters are authenticated by logging into the system. They will be prompted to enter their self-defined usernames and encrypted passwords for security purpose. As mentioned in the previous stage, users' passwords are not saved in the database. Only their hashed and salt values are saved. It is because Hashed-based Algorithms are one-way functions. They cannot be converted back to a plain text. To authenticate users, administrators are required to compare the ciphertext from user input with the ciphertext stored in the database. Figure 6 elaborates the process of user authentication in a format of the sequence diagram. Once a user has been identified as a registered voter and has successfully logged into the system, he will see a welcome screen, which states the user account status and a menu panel where a user can navigate through features offered, depending on the user level.

5.3. Voting Stage

This stage can only be completed by the voters. They are the only type of users who are eligible to cast the votes. In some of the E2E system, this stage is carried out by marking their chosen candidate(s) on the paper ballot, and then it needs to be scanned with the optical scanner or by using DRE machine to be submitted for the tally process. However, in eVote, 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 log on to the system and access the voting page. They can cast their votes by selecting their desired candidates for each category listed on that page. Besides that, the users could also review, reset and reselect candidates before they submit their votes. The voter's ballot is generated every time the chosen candidates are reviewed. During the ballot generation, F5 image steganography algorithm would be applied [29]. The F5 algorithm is implemented to secure data communication between the voter and the election server, even before the ballot is cast. The voter's chosen candidates would be encrypted in a stego-image format as a camouflage for their ballot.

The attackers would not have any idea that the client is sending their ballot to the server in a jpeg file format image. This ballot will, later on, be sent over to the tally server. Once received by the server, the ballot package would be decrypted to reveal the candidate names hidden on the ballot before being encrypted again with visual cryptography as an additional security level to earn voter's direct trust by providing vote receipt. This ensures that voters' votes have been collected as cast.

The decrypted stego-image (ballot) would be encrypted using a visual cryptography technique by splitting the vote into a number of shares. In this system, the shares would be limited to two. The stand-alone share would not reveal any information to anyone, but once the shares are overlaid (combined) using a visual cryptography decryption algorithm, the voter's casted vote value would be revealed. Basically, each voter would be given one layer (share) of the image as the receipt which will be sent to the respective email account, while the other separated layer of the vote would be kept or saved by the administrator for ballot counting purposes as well as to hide the relation of each voter with the ballot. Therefore, the voters would still be able to verify their votes and will have a better trust and confidence in the system. In the eVote system, the voting stage process is finished when the voting summary page is shown.

5.4. Tallying Stage

The tallying stage follows the voting stage. After the votes are cast, the ballot is 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 the help from system administrators. Each polling officer 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. The system administrators generate these keys by utilising UUID. Figure 7 elaborates this process in a sequence diagram. To access the tally list, polling officers must perform the 'de-

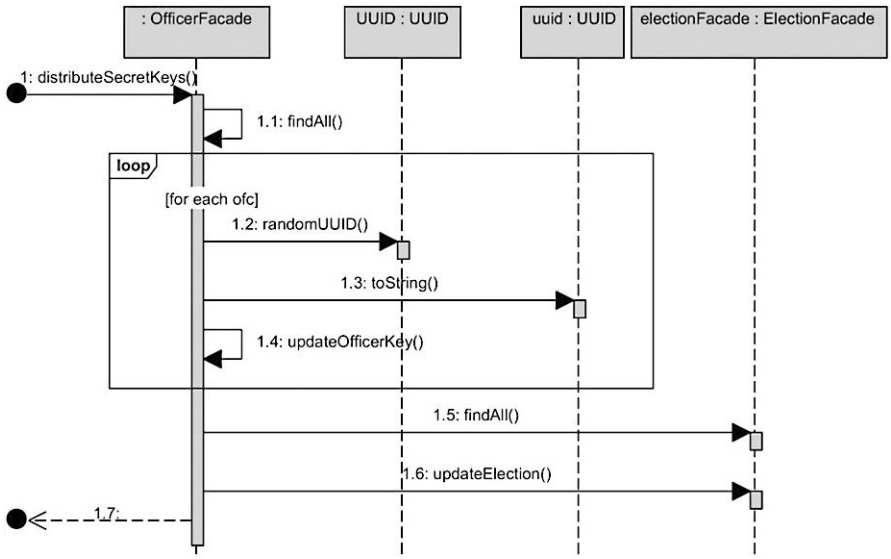

Figure 7. Sequence diagram of secret keys distribution process by the system administrator.

cryption' process by merging their secret keys. This method is called the threshold decryption cryptosystem [30]. Only after each of the polling officers has submitted their secret keys, the tally result 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 votes.

6. User Acceptance Testing of the Voting System

The users' acceptance was measured by using Davis' Technology Acceptance Model (TAM). For the questionnaire survey, 15 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 knowledge. The users were also chosen based on the minimum voting age requirement by the Malaysian law. Each of the participants was required to complete a set of voter's tasks assigned to them and to fill in a questionnaire in not more than thirty minutes. This test was conducted to consider not only technical factors to support the system's performance, but also to consider the behavioural factors of the users. To understand user acceptance level of the system, the Technology Acceptance Model [31] was used in this work. Such a model can provide a robust indication of the eVote's user acceptance. Davis claimed that these are two distinct cognitive appraisals of users' attitudes. They are the design features and the affective response to the system. In this test, two types of the design features proposed by Davis were applied to examine the user acceptance level. Those design features include perceived usefulness (extrinsic motivator) and perceived ease of use (intrinsic motivator). Perceived usefulness is the degree to which an individual believes that a system would enhance his or her job performance. On the other hand, perceived ease of use is the degree to which a user believes that the use of a system would require less effort compared with another system. Figure 8 shows the result collected based on the evaluation of perceiving the ease of use. Based on the evaluation conducted, the eVote's perceived usefulness aspect is shown in Figure 8. The 15 participants mostly preferred to cast their votes by using a remote E2E voting system, compared to casting their votes in the polling booth in a traditional voting system. As per the participants, the implementation of vote receipt is more reliable and offers more assurance to them, compared to the implementation of indelible ink used in the traditional voting system. There are many ways counterfeit votes can be cast using indelible ink due to the involvement of many parties in its

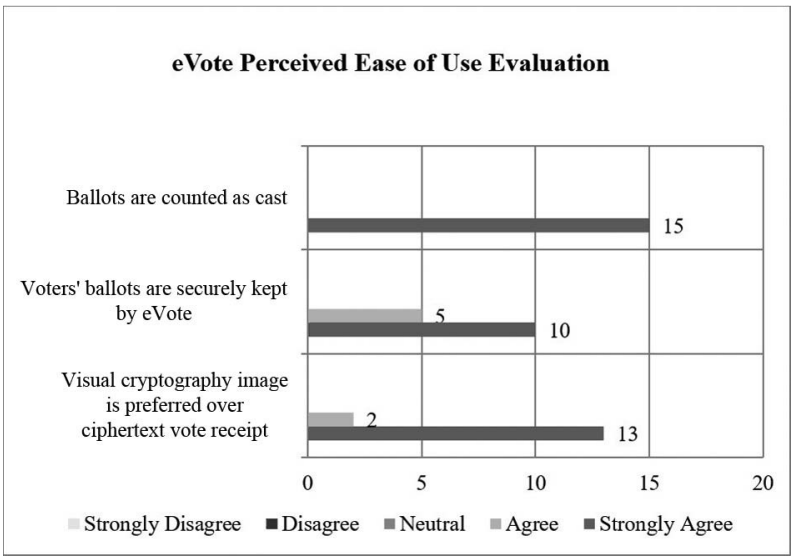

Figure 8. The eVote perceived ease of use evaluation.

implementation. However, the implementation of vote receipt only requires the involvement of system administrators and polling officers who are assumed trustworthy. Thus, it can be noted that the perceived ease of use affects the outcome of the usefulness evaluation. Perceived usefulness itself is the core factor of user acceptance level over the system.

From the result displayed in Figure 9, the result of the eVote's perceived usefulness aspect was expected. The user acceptance level could also be identified by evaluating the user requirements gathered in the requirements phase of the software development. Each requirement was grouped based on their characteristics and analysed as follows.

6.1. Mobility and Convenience

This is one of the aspects that an E2E system could not provide. Many of them used one or more election technologies that forced the users

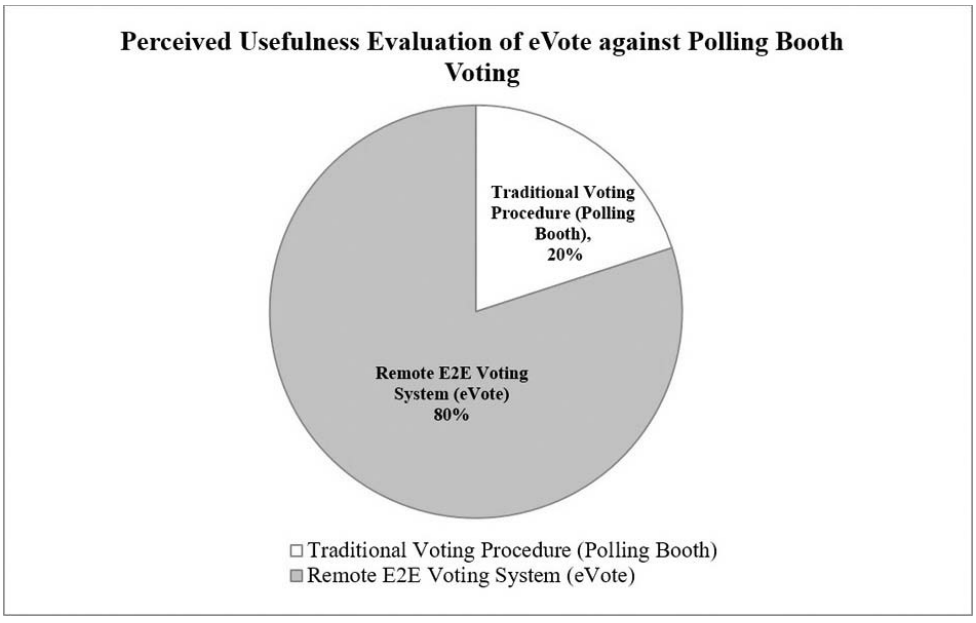

Figure 9. The eVote perceived usefulness evaluation.

to vote in a particular venue. The eVote offers a secure remote electronic voting system with vote verification feature which allows voters to vote anytime and anywhere, provided their PC is connected to the Internet.

6.2. Completeness

In a traditional voting, this requirement is simply catered by comparing the number of voters and the counted votes. This is merely time-consuming, and the chances of human-errors are high. On the other hand, the E2E online voting system could complete this task within seconds. The polling officers and system administrators are responsible for monitoring the ballots and the overall tallying process, including the tally result. However, vote coercion can still occur. The eVote voting system reduces this by eliminating direct access to the database without the consent of multiple parties. The users can only access the database server from the system. It ensures that the completeness of eVote voting system would not be violated.

6.3. Privacy, Soundness, Verifiability and Robustness

The deliverance of these aspects is ensured in the eVote voting system's design by implementing image steganography. The vote transmission during voting process in eVote is secured beforehand by encoding the cast ballot with F5 image steganography algorithm. As the privacy of the votes could be guaranteed, the soundness and verifiability could be carried out as well because no one can obtain any information on the votes except the voters themselves.

6.4. Incoercibility and Receipt-freeness

As visual cryptography receipt is included in eVote's system architecture, these two requirements are supported. It is used in conjunction with visual cryptography to ensure the integrity of the election and to prevent improper influence in the election process. The receipt is secure and in the format of visual cryptography image share. Untrustworthy voters are not able to prove to a coercer that they have voted in a particular way. Thus, incoercibility and receipt-freeness are supported in the eVote system.

6.5. Fairness

In the tallying stage, the threshold decryption scheme was applied to keep the vote secure. The read-only tally result is only accessible to polling officers and system administrators after the election has ended. The distributed secret receipts should be submitted by all the polling officers to the election server. Because of this, no one can gain any information regarding the tally result before the election ends.

A comparison of the E2E voting system components is given in Table 1 for different online voting systems. The eVote is comparable to many of them and better in some features.

7. Conclusion

As we know, voting systems are classified into two types based on their ballot mechanism, namely paper-based E2E voting systems and electronic E2E voting systems. As a receipt-based voting system, both voting systems allow their voters to verify the accuracy of their votes to ensure that they have been collected as cast and counted as collected by using the vote receipt. Thus, incoercibility is offered in such voting systems. This however, creates other vulnerabilities in the paper-based E2E voting system. They are caused by the dependency on a few aspects, for example, possible errors made from the integration of the voting system with an additional external hardware (e.g. DRE machine) and human errors during the vote tabulation. On the other hand, even though the implementation of vote receipt in the electronic E2E voting systems eliminates this problem, the vote receipt feature offered must be practical and convenient. It must meet the basic user-requirements, on top of the security requirements. However, these aspects are sometimes overlooked. These issues have been addressed in the eVote voting system development. The eVote is developed based on the password hashed-based scheme, visual cryptography as adopted from secret-ballot receipt scheme, threshold decryption cryptosystem and F5 image steganography for security, which do not have high computational cost.

The contribution of this paper lies in the simplicity and user-friendliness it offers without

Table 1. Comparison of E2E voting system components with different voting systems.

Features	Helios	Scantegrity	Prêt a Voter	RIES	eVote
Universal Verifiability	Yes	Yes	Yes	Yes	Yes
Individual Verifiability	Yes	Yes	Yes	Yes	Yes
Mobility	Yes	No	No	Yes	Yes
External Hardware	No	Yes	Yes	Yes	No
Main Implemented Technology	Mix-net Scheme and Threshold Decryption	Optical Scan and Anonymity Network	Visual Cryptography and Chaum's Secret-ballot Receipt	Cryptographic Hash Function	F5 Image Steganography and Visual Cryptography
Bulk Registration	Yes	No	No	No	Yes
Administrator	Yes	Yes (referred to as Scantegrity team)	Yes (referred to as election authorities)	Yes (referred to as TTPI)	Yes
Voter	Yes	Yes	Yes	Yes (including the election board)	Yes
Election Trustees or Officer	Yes	Yes	Yes (referred to as auditors and help organisations)	N/A	Yes

compromising the system security and usability. The vote receipt in eVote was implemented in visual cryptography image format. This type of vote receipt format is more practical and convenient than the vote receipt in ciphertext format. The solution has no effect on the system security. This system has been designed and developed to provide a more secure voting system compared to other E2E verifiable voting systems. The system is also scalable if a high end and security hardened election server is installed in a more professional environment. Then it can be used for large-scale elections. Though we do not claim that the system is fully secure against all forms of attacks, it can provide a reasonably secure environment for online voting.

References

[1] S. T. Ali and J. Murray, "An Overview of End-to-End Verifiable Voting Systems", Voting – Design, Analysis and Deployment, Edited by H. Feng and P. Y. A. Ryan, Taylor & Francis Group, *CRC Press*, pp. 171–218, 2016.

[2] A. Kiayias *et al.*, "An Authentication and Ballot Layout Attack Against and Optical Scan Voting Terminal", *Proceedings of the USENIX workshop on accurate electronic voting technology*, ACM Digital Library, 2007.

[3] R. Kofler *et al.*, "Electronic Voting: Algorithmic and Implementation Issues", in *System Sciences Proceedings of the 36th Annual Hawaii International Conference*, IEEE Computer Society, Washington DC, USA, 2003, pp. 7. http://dx.doi.org/10.1109/HICSS.2003.1174319

[4] P. Y. A Ryan *et al.*, "Prêt à Voter: a Voter-Verifiable Voting System", *IEEE Transactions on Information Forensic and Security*, vol. 4, no. 4, pp. 662–673, 2009. http://dx.doi.org/ 10.1109/TIFS.2009.2033233

[5] P. Vora, "David Chaum's Voter Verification Using Encrypted Paper Receipts", Cryptology ePrint Archive, Report 2005/050, 2005.

[6] D. A. Gritzalis, "Principles and Requirements for a Secure E-Voting System", *Computers & Security*, vol. 21, no. 6, pp. 539–556, 2002. http://dx.doi.org/10.1016/S0167-4048(02)01014-3

[7] J. Benaloh, "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, 2006.

[8] S. Chandramathi *et al.*, "An overview of Visual Cryptography", *International Journal of Computational Intelligence Techniques*, vol. 1, issue 1, pp. 32–37, 2012.

[9] D. Chaum, "Secret-Ballot Receipts: True Voter-Verifiable Elections", *IEEE Security and Privacy*, vol. 2, no. 1, pp. 38–47, 2004. http://dx.doi.org/10.1109/MSECP.2004.1264852

[10] N. Provos and P. Honeyman, "Hide and Seek: an Introduction to Steganography", *IEEE Security & Privacy*, vol. 1, no. 3, pp. 32–44, 2003. http://dx.doi.org/10.1109/MSECP.2003.1203220

[11] J. Z. Hong and J. T. Hong, "A Novel Image Steganography Algorithm Against Statistical Analysis", *Proceedings of machine and cybernetics 2007 international conference*, vol. 7, pp. 3884–3888, 2007. http://dx.doi.org/ 10.1109/ICMLC.2007.4370824

[12] P. Bateman, "Image Steganography and Steganalysis", Master's thesis, Faculty of Engineering and Physical Sciences, University of Surrey, 2008.

[13] T. Morkel *et al.*, "An Overview of Image Steganography", in *Proceedings of the Fifth Annual Information Security South Africa Conference in Sandton*, South Africa, pp. 1–11, 2005.

[14] E. Hubbers *et al.*, "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, 2005. http://dx.doi.org/10.1109/COMPSAC.2005.132

[15] R. Carback *et al.*, "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, 2010.

[16] D. Chaum *et al.*, "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, no. 14, 2008.

[17] B. Adida, "Helios: Web-based Open-Audit Voting", in *Proceedings of the 17th Conference on Security Symposium*, USENIX Association, Berkeley, USA, pp. 335–348, 2008.

[18] M. Naor and A. Shamir, "Visual Cryptography, Advances in Cryptology – EUROCRYPT 1994", Workshop on the Theory and Application of Cryptographic Techniques, in *Proceedings of Lecture Notes in Computer Science*, Springer-Verlag, pp. 112, 1994. http://dx.doi.org/10.1007/BFb0053419

[19] W. M. Santos *et al.*, "Toward Coercion-Resistant End-to-End Verifiable Electronic Voting Systems", *12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, Melbourne, pp. 1696–1703, 2013. http://dx.doi.org/10.1109/TrustCom.2013.278

[20] N. Chondros *et al.*, "D-DEMOS: A Distributed, End-to-End Verifiable, Internet Voting System", *IEEE 36th International Conference on Distributed Computing Systems (ICDCS)*, Nara, pp. 711–720, 2016. http://dx.doi.org/10.1109/ICDCS.2016.56

[21] N. H. Sultan *et al.*, "PairVoting: A Secure Online Voting Scheme Using Pairing-Based Cryptography and Fuzzy Extractor", *IEEE International Conference on Advanced Networks and Telecommuncations Systems (ANTS)*, Kolkata, pp. 1–6, 2015. http://dx.doi.org/10.1109/ANTS.2015.7413634

[22] J. L. Tornos *et al.*, "An eVoting Platform for QoE Evaluation", *IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, Ghent, pp. 1346–1351, 2013.

[23] M. Yi *et al.*, "Efficient Security Sequencing Problem over Insecure Channel based on Homomorphic Encryption", *China Communications*, vol. 13, no. 9, pp. 195–202, 2016. http://dx.doi.org/10.1109/CC.2016.7582311

[24] P. S. Naidu and R. Kharat, "Secure Authentication in Online Voting System Using Multiple Image Secret Sharing", in Security in Computing and Communications. SSCC 2016, (P. Mueller, Eds.), *Communications in Computer and Information Science*, vol. 625, Springer, Singapore, 2016. http://dx.doi.org/10.1007/978-981-10-2738-3_29

[25] A. Goncalves, "Beginning Java EE 6 with GlassFish 3", 2nd edition, Apress, USA, 2010.

[26] J. W. Satzinger *et al.*, "Systems Analysis & Design in a Changing World", 5th edition, Cengage Learning, 2008.

[27] L. Rura *et al.*, "Analysis of Image Steganography Techniques in Secure Online Voting", in *Proceedings of IEEE International Conference on Computer Science and Network Technology (ICCSNT)*, pp.120–124, 2011. http://dx.doi.org/10.1109/ICCSNT.2011.6181922

[28] L. Rura *et al.*, "Online Voting Verification with Cryptography and Steganography Approaches", in *Proceedings of IEEE International Conference on Computer Science and Network Technology (ICCSNT)*, pp.125–129, 2011. http://dx.doi.org/10.1109/ICCSNT.2011.6181923

[29] A. Westfeld, "F5 – A Steganographic Algorithm", in *Proceedings of the 4th International Workshop on Information Hiding (IHW '01)*, Springer-Verlag, London, UK, pp. 289–302, 2001. http://dx.doi.org/ 10.1007/3-540-45496-9_21

[30] A. Shamir, "How to Share a Secret", *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.
http://dx.doi.org/10.1145/359168.359176

[31] F. D. Davis, "Perceived Usefulness, Perceived Ease of Use and User Acceptance of Information Technology", *MIS Quarterly*, vol. 13, no. 3, pp. 319–340, 1989.
http://dx.doi.org/10.2307/249008

Received: February 2016
Revised: March 2017
Accepted: March 2017

Contact addresses:
Lauretha Rura
Faculty of Engineering, Computing and Science
Swinburne University of Technology (Sarawak Campus)
Kuching, Malaysia
e-mail: lrura@swinburne.edu.my

Biju Issac
School of Computing
Teesside University
Middlesbrough, UK
e-mail: bissac@ieee.org

Manas Kumar Haldar
Faculty of Engineering, Computing and Science
Swinburne University of Technology (Sarawak Campus)
Kuching, Malaysia
e-mail: mhaldar@swinburne.edu.my

Lauretha Rura received her Master of Science (by research) from Swinburne University of Technology (Sarawak Campus), Malaysia. Her main research topic is the improvement of E-voting systems, as well as E2E verifiable voting systems.

Biju Issac is working at Teesside University as an academic staff member. He holds a PhD degree in Networking and Mobile Communications, along with the MCA (Master of Computer Applications) and BE in Electronics and Communications Engineering. Dr Issac is an active researcher and has authored more than 80 refereed conference papers, journal papers and book chapters.

Manas Kumar Haldar has obtained his PhD as a 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.