


Security, usability, and biometric authentication scheme for electronic voting using multiple keys

International Journal of Distributed
Sensor Networks
2020, Vol. 16(7)
© The Author(s) 2020
DOI: 10.1177/1550147720944025
journals.sagepub.com/home/dsn


Masood Ahmad¹, Ateeq Ur Rehman¹ , Nighat Ayub², MD Alshehri³,
Muazzam A Khan⁴ , Abdul Hameed⁵ and Halil Yetgin^{6,7}

Abstract

We propose electronic voting authentication scheme, which is a key management mechanism for electronic voting system intended to limit the number of attacks on a polling station and strengthen the security control. The motivation is to diversify security requirements of messages exchanged between polling stations. There are different types of messages exchanged between polling stations and each type of message has different security needs. A security mechanism developed on the basis of a single key is not enough to ensure the diverse security needs of voting network. In electronic voting authentication scheme, every polling station is responsible to support three different types of keys. These are global key, pairwise key, and individual key. The global keys are public keys shared with all polling stations in the voting network. The pairwise key can be used for communication with polling stations. Individual keys will be used for communication with the server. To ensure authentication of local broadcast, electronic voting authentication scheme uses one-way key chains in a well-organized way. The support of source authentication is a visible advantage of this scheme. We examine the authentication of electronic voting authentication scheme on numerous attack models. The measurement demonstrates that electronic voting authentication scheme is very operative in protecting against numerous elegant attacks such as wormhole attack, Sybil attack, and HELLO Flood attack. The proposed system is evaluated and the results demonstrate that the proposed system is practical and secure as compared to the direct recording electronic and manual systems.

Keywords

Electronic voting, security, authentication, attacks, key distribution, key management

Date received: 6 January 2020; accepted: 22 June 2020

Handling Editor: José Camacho

¹Department of Computer Science, Abdul Wali Khan University, Mardan, Pakistan

²Department of Computing and Technology, Iqra University, Islamabad, Pakistan

³College of Computers and IT, Taif University, Taif, Saudi Arabia

⁴Department of Computer Science, Quaid-i-Azam University, Islamabad

⁵Department of Higher Education Archives and Libraries, Khyber Pakhtunkhwa, Pakistan

⁶Department of Communication Systems, Jozef Stefan Institute, Ljubljana, Slovenia

⁷Department of Electrical and Electronics Engineering, Bitlis Eren University, Bitlis, Turkey

Corresponding authors:

Ateeq Ur Rehman, Department of Computer Science, Abdul Wali Khan University, Mardan 23200, Pakistan.

Email: ateeq@awkum.edu.pk

Muazzam A Khan, Computer Science, Quaid-i-Azam University, Islamabad.

Email: muazzam.khattak@qau.edu.pk



Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (<https://creativecommons.org/licenses/by/4.0/>) which permits any use, reproduction and distribution of the work

without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (<https://us.sagepub.com/en-us/nam/open-access-at-sage>).

Introduction

In almost all democratic countries, voting is practiced regularly. The most common method used in elections is the voting based on paper.¹ The paper ballots are used and the voters cast their vote physically. The counting mechanism in these elections is manual. The voter's intent is estimated from a physical ballot. The results are tabulated manually after the interpretation and reading of physical ballots.

The paper-based voting scheme may be used for recounts. The recounts are applicable where automated or mechanical counting systems are used. While this type of voting systems is dominated by most of the election systems conducted in different countries in terms of problems arose during the election. The reports of fraud or cheat are frequently highlighted in these elections. The reports may be due to voting itself or errors in counting procedure. Many problems were documented in the previous years about the election process.² The time required for the election process and the result announcement are other limitations of the manual voting systems.

Electronic voting (e-voting) technology may offer advantages over the existing manual voting scheme.³ The speed of both counting and vote casting is increased. The access to disable voters is granted without visiting the polling station (PS). The e-voting is conducted in many countries of the world, including Europe, Australia, and United States. In Kuwait, an attempt was made in 2005 parliamentary elections to use e-voting, but the result was not satisfactory due to a number of problems.⁴

However, conducting elections through e-voting has a number of advantages over manual voting scheme, but the issues related to usability, security, and credibility still exist. In the meantime, the e-voting is still in its infancy and comparatively immature, which can be improved from the lessons learned and experiences gained through previous e-voting attempts. For example, in United States presidential elections conducted in 2000, a number of issues in e-voting were reported.⁵

The objective of this research is to convey the experience of e-voting deployed at the University of Malakand regarding usability, security, efficiency, and satisfaction. The potential voters are the students of the university. The objective of this experiment was the elimination of a large number of wrong votes in addition to other objectives. The votes may be wrong because of execution errors which the voters made during vote cast. Execution errors arise when the intentions of voter were correct before the vote. In other words, the action was not as intended. This occurs because of unsuitable or unclear designs. The voters are authenticated by the biometric verification system, unlike manual authentication. The wrong ballots are

very minimal as the design of the proposed system is very simple and anyone can understand the voting process very quickly. The voters are guided through easy instructions in Urdu language during the vote casting process. A comprehensive questionnaire is designed to address the usability issues. The voter's feedback regarding the usability of the system is recorded and the data are then analyzed. Due to the strong authentication mechanism and easy user interface, the results show that the proposed system is more secure and user friendly as compared to the existing systems. Security of the votes is achieved via multiple keys during the communication of PSs and a base station.

Electronic voting systems are very much attracted to the hackers in recent years.⁶ Therefore, for the proper functioning of many e-voting systems, the security schemes for authentication⁷ and confidentiality confirmation are very difficult to design. Due to the need and interest of people in the democracy,⁸ ensuring security is particularly challenging in e-voting process. The use of asymmetric cryptography⁹ in e-voting is impractical. Therefore, key management mechanisms for e-voting systems are established on the basis of symmetric key cryptography.

An important concern that needs careful attention is the process used for key management schemes for symmetric shared keys.¹⁰ At present, pre-deployed keying mechanism is the most commonly used method for bootstrapping secret keys in e-voting systems, where the keys are loaded into PSs prior to deployment.

The degree of key sharing¹¹ among PSs in the network is one of the essential design considerations for security schemes based on symmetric keys. For authentication and data encryption, network wide keys can be used due to less communication requirements among PSs for creating additional keys. The problem is of key compromise on a single PS results in global key (GK). However, we have a key sharing method in which all secure communication is based on a key pair shared between two PSs.

This approach is ideal from the security point of view in scenarios where a PS compromise does not mean that the key in other PSs have been revealed. In this approach, a unique key will be used when a PS wants to communicate with other PSs.

To encourage passive contribution, it is important that the intermediary PS is able to verify or decrypt a secure message communicated between the two other PSs. If multiple PSs share the keys intended for authentication and encryption, passive involvement of secure messages is possible. If message authentication or encryption is performed with a pairwise shared key, it efficiently prevents passive participation in e-voting systems. The major contributions of this article are outlined as follows:

1. A new e-voting system is developed and its usability is evaluated. The key characteristic is the biometric confirmation of the voter and vote casting. In comparison to dual vote, the proposed e-voting system needs no training overhead before the election.
2. To evaluate the reliability, vote on paper-based, direct recording electronic (DRE), and proposed e-voting system was cast. The results show that the proposed electronic voting authentication scheme (EVAS) outperforms DRE and manual voting.
3. We propose EVAS, a key management scheme for e-voting systems that provide security features of a pairwise key (PK) sharing mechanism. The impact of a PS compromise on the PSs in their neighborhood can be minimized with the EVAS key management scheme. EVAS contains provision for multiple key management schemes. Due to the fact that the messages that are interchanged among PSs have different security needs and a security scheme based on a single key is not suitable. More precisely, EVAS establishes three different types of key that can be deployed in each PS. These are GK, PK, and individual key (IK).
4. The benefit of this approach is the minimization of server involvement in the establishment of the keys.
5. Utilizing the concept of one-way key chains, EVAS also contains a well-organized scheme for the authentication of local broadcast. A noticeable benefit of the authentication mechanism is the support for source authentication (unlike a mechanism where authentication is provided via GK) without inhibiting passive involvement (unlike a mechanism where authentication is provided through pairwise shared key).

The rest of article is structured as follows. The related work and critical analyses are discussed in section “Related work.” The proposed scheme EVAS is presented in section “Proposed e-voting system.” The authentication approach for local broadcast is discussed in section “EVAS,” while the report about EVAS security analysis is presented in section “Security analysis.” Finally, the article is concluded in section “Conclusion.”

Related work

Electronic voting has gained attention over the years, but till now limited number of voting machines have been used for real elections.¹² Security and cryptographic voting protocols are rarely used by election officials. The main focus of this research is on making usable and verifiable remote e-voting systems, which have been neglected in former elections. The election was set up using the Helios voting system. They performed a usability analysis of Helios version 3.0 using some additional parameters.¹² In addition to their previous work, this article provides an experimental study. In this experimental study, users are selected from both technical and non-technical backgrounds. No advanced knowledge is provided to the users about the verifiable e-voting system. The technical users raised a complaint that the information is not sufficient for voting purpose, while the non-technical users argued that the system is not secure and that the verification process requires additional effort. Previously used user interfaces are further improved for this research work. Table 1 summarizes features of different voting systems.

The authors of Al-Ibrahim and Al-Ostad¹³ emphasize on the user interface and on the design issues related to e-voting system, which is evaluated on the basis of usability parameters. Usability is achieved by designing an e-voting machine. The election is

Table 1. Usability and security of different voting systems.

Reference	Usability	Security	Verifiability	User authentication	Voter anonymity	Voter privacy	Cost
Tretyakov et al. ¹¹	Easy	Moderate	No	No	No	No	Low
Karayumak et al. ¹²	Complex	Moderate	Strong	By email	No	No	Low
Al-Ibrahim and Al-Ostad ¹³	Easy	No	No	No	No	No	High
Karayumak et al. ¹⁴	Easy	No	Weak	No	No	Yes	High
Mac Namara et al. ¹⁵	Complex	No	No	No	Yes	No	Low
Uzunay and Bicakci ¹⁶	Complex	Not sure	No	No	No	No	Moderate
Bruns et al. ¹⁷	Easy	Moderate	No	Yes	No	No	Moderate
Campbell et al. ¹⁸	Easy	No	No	Yes	No	No	Low
Khairnar and Kharat ¹⁹	Moderate	Moderate	No	Yes	Yes	Yes	Low
Walake and Chavan ²⁰	Moderate	Moderate	No	Yes	No	No	Low
Al-Anie et al. ²¹	Complex	Moderate	No	Yes	No	No	Low
Binu et al. ²²	Moderate	Moderate	No	Yes	No	No	Low

conducted through this voting machine and the results were analyzed based on usability and user interface designing. The main goal of this voting machine is to eliminate invalid ballots, enable correct ballot accounting, and support active result revelation. The authors of Al-Ibrahim and Al-Ostad¹³ also discuss security but to a limited extent. Authentication was only performed manually, which is in contrast to the nature of e-voting. However, anonymity and privacy were not considered in their proposed system, where voter information was printed on the ballot so anyone could know who cast vote to whom.

In Karayumak et al.,¹⁴ the analysis of Helios open source end-to-end (E2E) electronic system is performed in terms of usability and security, and e-voting experts follow cognitive walk-through approach. According to the authors of Karayumak et al.,¹⁴ usability and verifiability of Helios need improvements, and only after rectifying the Helios, it may be used in high level elections. New interfaces for Helios are proposed to improve the usability of E2E verifiable e-system. Improved interfaces have been proposed for Helios, but there are some bottlenecks in the system. The main drawback of the system is that no back buttons are provided and that a very limited number of voters would be able to cast votes because of the complexity of the system. Verifiability is also an issue of great concern.

User authentication is conducted via email whereas not every citizen has an email address in country's election system and some even do not know how email works. The overall system is complex and difficult to understand. It is also very difficult to use in a country-wide election because not every citizen has access to Internet and computer. The terms utilized in this prototype are also confusing, such as audit, encrypt as some of the citizens may not understand these terms. The interfaces used in the system were not attractive and user friendly.

The purpose of Mac Namara et al.¹⁵ is to develop a dual voting system named "Just Like Paper (JLP)." In their proposed system, the voters cast their vote without any hesitation using an electronic pen. JLP is not very broad in the sense that it only focuses on the design of user interface and on the usability aspects of e-voting systems. In contrast to the JLP, the electronic voting classification structure (EVCS) emphasizes on the creation of a worldwide applicable language for e-voting system. In addition, the authors of Mac Namara et al.¹⁵ classify 26 different e-voting systems, where they differentiates between conventional voting system and e-voting system. The authors analyze the main differences between features of these systems. The main drawback of this system is that the utilized electronic pen for casting the vote on an e-voting machine introduces additional costs. The use of this pen is also

cumbersome as special training may be required. The voter verification process is also not explained in this model.

The authors of Uzunay and Bicakci¹⁶ explain the ThreeBallot system, which is an E2E verifiable e-voting system in the absence of any security mechanism. A system without a proper security mechanism has not only some usability but also security flaws. Therefore, the authors of Uzunay and Bicakci¹⁶ proposed Trusted3Ballot, which is a trusted computing-based voting scheme with three ballots addressing the security and usability related problems. The system is very complex. Extra overhead was involved in processing the operations of three Ballot system. Special kernel is required for the voting system and it requires specialized software that increases the cost in terms of software development. The proposed system cannot be implemented on existing operating systems and infrastructure and it uses bootable CDs to load the operating system and voting software which may interrupt the whole election process as the CD may become unusable after several use. Bar code readers are used for authentication of the voter and a fake voter may cast the vote on behalf of the original voter. The key generation process is carried out in front of participants that may increase the chance to reveal the keys to hackers.

The authors of Desmedt and Erotokritou²³ have proposed two protocols, namely, the mod 10 Internet voting and the permutation Internet voting. The authors claim that the proposed method provides security even in the presence of malware. Mod 10 Internet voting protocol is actually a code voting protocol and is more user friendly. Users from different races are evaluated for permutation-based protocols through different experiments and it is demonstrated that these protocols are user friendly and race-neutral. Only a fraction of the voters is computer literate; thus, it is difficult to deploy such a system, where the code generation is an additional task of the voters.

In the proposal of Bruns et al.,¹⁷ automated tools are used to ensure the non-interference in elections. The implementation is carried out in Java and the programs are secured from non-interference. As the automated tools suffer from false positives, the theorem prover is then needed to ensure the non-interference, which is mostly a time-consuming process. Moreover, in Bruns et al.,¹⁷ an automated tool like Joana is used for checking non-interference and the automatic theorem prover is used for further analysis. A slicing method is used for program slicing, where part of the program that does not influence the final result is discarded from verification process. Hence, debugging process became simpler. The authors of Bruns et al.¹⁷ attempted for a secure system; however, the implementation details of the system and the voting mechanism as well as the authentication of the voters were not thoroughly discussed.

To further extend the studies, we also reviewed the security mechanisms proposed in the literature and outlined their objectives and elaborated on their findings related to the system for handheld mobile devices.¹⁸

Khairnar and Kharat¹⁹ proposed a secure online system for vote casting where authentication is performed using registration through thumb impression. Only the voter with registered thumb is capable of casting a vote. Through thumb impressions, all the information is sent to the server, which then allocates password and ID to the voter after the login. After casting the vote, vote is encrypted and stored in a vote recording server. Security of the voter is the main objective of this system and it also encrypts the votes through homomorphic encryption and blind signatures. The system is user friendly and is secure enough.

The authors of Walake and Chavan²⁰ focus their attention on the online voting system and authentication mechanisms, where the authentication is provided through Shamir's secret sharing system. There are two phases of this system including the registration phase and the authentication phase. Registration of voter and candidate is mandatory within this system. The system will generate passwords after the registration process. The passwords are then divided into shares by the trusted center. Central authority and voter get hold of the share. In authentication process, secret sharing is an important phenomenon to secure the authentication process. For authentication, both the voter and the authority shares are required to produce the original secret. Using this approach, vote casting will only be allowed for the valid users. In their proposed system, there is no security or any encryption method discussed for the vote casting procedure. Security discussions are only limited to the authentication process.

Al-Anie et al.²¹ established e-voting protocols based on public key encryption, which enabled the voters to cast their votes from their own computers. There are three phases of this system, that is., the authentication and registration phase (the involvement of civil status and passport is mandatory in this phase), the voting phase, where the voter's information is encrypted through public key encryption and submitted to the government election server through the network, and the mobile company verification phase as the mobile phone company will allocate a personal identification number (PIN) to the voter. Then, the election server administrator will decipher the encrypted information using the Rivest-Shamir-Adleman (RSA) private key to sort out the final result. However, a lot of overhead is involved in this system, as the system requires a standard mobile phone with a strong Internet connection, personal computer, and also the use of voting website. It is also easy to break the PIN allocated by the mobile phone company to the voter.

Binu et al.²² proposed a secure and efficient e-voting scheme, which is purely based on the secret sharing homomorphism for efficient voting and verification. They were inspired by the Shamir to store the secret keys in different locations. Their proposed scheme provides a better way of encoding and decoding of votes. A vote is considered as a secret in their system and using the keys to access the encoded vote makes it reliable and no one can access to it with malign intentions. However, their proposed scheme requires a secure channel and can be solely used in a moderate system that does not deal with too many votes (shares).

Zwierko and Kotulski²⁴ proposed an agent based scheme for secure e-voting and can be used both in mobile and geographically fixed areas. Their proposed scheme is based on cryptography and secure secret sharing. The advantage of the scheme is that users do not have to do computations. Their proposed system is suitable for almost all kind of elections. An agent installed on the mobile phones helps voters to cast the votes, thus allowing the voters to be geographically independent, where the cryptographic algorithms ensure the security.

Proposed e-voting system

The main objectives of the proposed e-voting system are

- Easy and reliable authentication using biometric devices.
- To prevent rigging in elections by the polling agents and polling staff.
- To get rid of paper ballots, so the polling staff will be minimized.
- To support free and fair elections.
- To assist in announcing election results quickly.
- To mitigate mistakes encountered during vote accounting.
- Reduced number of "invalid ballots."

The architecture of the proposed e-voting system contains a base station and a number of PSs (e-voting devices). The base station is a kind of server that manages the e-voting system. The PSs are connected to the base station via a private local network, where base station monitors and manages all the activities of PSs by opening and closing sessions for voting. The votes (e-votes) from PSs are transmitted and stored in a database of the base station. The process of the electronic election is shown in Algorithm 1.

In our experiment, the voter data are stored before starting an election, where the voters are asked to enter thumb impressions via a biometric device and then the data are recorded in a database. The stored data are

Algorithm 1. Pseudocode of algorithm e-voting

```

1: procedure e-voting
2: Input: blank database
3: Output: result
4: call procedure pre-election (database)
5: while (voters !=null or election-time !=end) do
6:   call procedure voter-authentication(thumb-impression)
7:   call procedure vote-cast (voter)
8:   call procedure evoting-EVAS (vote)
9:   call procedure store-vote (vote)
10: end while
11: call procedure announce-result (database, decryption-key)
12: end procedure

```

Algorithm 2. Pseudocode of algorithm pre-election

```

1: procedure pre-election
2: Input: blank database
3: Output: voter-database // having voter information
4: while ( voter != null) do
5:   Ask user to place figure on biometric device and get
the thumb impression
6:   m-bimage1 = image IO. read (new figure)
7:   m-fingure1 = set Finger Print Image (m-bimage1)
8:   m-image1 = m-fingure1. get Finger Print Image Detail()
// store register voter data in database
9:   insert into voterTable (NIC, m-image1)
10: end while
11: return database
12: end procedure

```

used to authenticate the voter before casting their vote. For practical implementation of e-voting system, the data may be obtained from country databases like the National Database and Registration Authority (NADRA) database in Pakistan may be used for the purpose of authentication. However, we do not have access to such data and maintained our own database locally. The process of storing voter information before election is elaborated in Algorithm 2.

When a voter wants to cast their vote, he or she must pass through three consecutive steps in order to successfully cast their vote. These steps are authentication, voting, and casting.

The main goal of the first step is to ensure the voter's identity. The authentication is performed by the system through biometric devices that check the identification of the voter and ensure his or her eligibility by comparing with the list of voters recorded in the database of the respective server. The thumb of the voters is actually compared with the thumb stored in the database. Algorithm 3 is utilized for ensuring the authenticity of the voter.

Once the thumb is matched, the information of the voters is displayed on the screen including name,

Algorithm 3: Pseudo code of algorithm voter-authentication

```

1: procedure voter-authentication
2: Input: database, voter data
3: Output: Boolean value // true for authentic voter and
false for voter having no information
4: Ask user to place figure on biometric device and get the
thumb impression
5: nic=nicinput.inputNicNumber()
6: ImageTest=imageResize(nic)
7: m-bimage1=image IO. read (ImageTest)
8: m-fingure1=set Finger Print Image (m-bimage1)
9: m-fingure2=get Image from Database
10: mper=m-fingure1.Match (fingure1, fingure2)
11: if (mper greater than 50) then
12:   match=true
13: else
14:   match=false
15: end if
16: return match
17: end procedure

```

Algorithm 4: Pseudo code of algorithm cast-vote

```

1: procedure cast-vote
2: Input: voter information
3: Output: vote
4: if (voter=true) then
5:   showBallot()
6:   select your choice
7:   Review choice
8:   while (choice=wrong) do
9:     showBallot()
10:    review choice again
11:    choice=true
12:   end while
13: end if
14: submit-vote()
15: return voter-choice
16: end procedure

```

father's name, vote number, PS name, and so on, as shown in Algorithm 4. Once authenticated, the voter is able to cast their vote.

In the voting stage, a session is started on the indicated PS for a specific period. The session expires when a voter casts their vote or the system remains idle for 60 s. When a user clicks on the next button in the information screen after verifying that the information displays on the screen is his or her own information, the next screen that must appear is the electronic ballot paper. The ballot paper is designed in a simple way which contains the candidate's name and election symbol in front of a radio button. The voter only needs to touch one radio button in front of his or her choice. The user can change the choice before submission with

Algorithm 5: Pseudo code of algorithm announce-result

```

1: procedure announce-result
2: Input: Database, Decryption-key
3: Output: election result
4:   while (votes !=null) do
5:     select vote from voteTable
6:     decryptVote(vote)
7:     for (j=1; j<=candidates; j++ ) do
8:       if (vote=candidateNames(j) then
9:         Candidate-votes[j] ++ //increment candidate
votes
10:      end if
11:    end for
12:  end while
13: return candidate-votes
14: end procedure

```

one click as only one radio button can be selected from the list of radio buttons.

The next step is vote casting. After the selection of a candidate, the voter needs to touch the “cast-vote” button. When the voter press “cast-vote” button, the vote data are transmitted to the base station. The vote data are encrypted in the PS using IK before being transmitted to the base station, which then stores the data in a database. After the final stage, a welcome screen appears again for the next voter. The voters cast their votes electronically in the order of which they cast their votes manually. Manual voting using paper ballot is done to cross check the results of the e-voting election. This shows that vote is cast for the candidate using e-voting machine to which the voter intended. With this method, the results of the wrong ballots are obtained.

The e-voting system discussed above was implemented at the University of Malakand, Pakistan. A touch screen laptop was deployed to conduct the election. Virtual machine was used to create the client server architecture. Before starting the counting process, the base station and all the ballot boxes were moved to the central library at the end of the Election Day. A program is designed that automatically counts the number of votes a candidate got, as shown in Algorithm 5. The program retrieves the vote data from the database stored in the base station. Hence, the counting program was activated for the results to be displayed on the screen, which contains information about the number of votes against the candidate name at the end of the election. The program performs data retrieval from the database and other statistics in milliseconds. The credibility of the proposed e- voting is verified. The credibility is assured in a way that the paper-based ballots were counted manually and the result of e-voting was cross checked, and both results were found identical in our experimentation.

Each voter was asked to fill a questionnaire after successfully casting their vote. In the questionnaire, a few questions were asked about the usability of the

system. A few other information was also recorded during the election, such as ballot completion time. In the following sections, the security mechanism of the proposed e-voting system and its results are presented.

EVAS

EVAS contains multiple key management scheme²⁵ that provides authentication²⁶ and confidentiality²⁷ in e-voting systems. The establishment of multiple keys can be best described after an overview of the different key management schemes proposed in the literature. The local broadcast scheme is also presented in section “Security analysis.”

Overview

To ensure communication and successful election, there are different types of data packets that need to be interchanged between PSs of the e-voting system. These packets can be categorized based on different criterion such as data packets, control packets, unicast packets, and broadcast packets. The category of a packet can change the security requirements of a packet. For some packets, the confidentiality is mandatory, whereas authentication is mandatory for all kinds of packets. For instance, the confidentiality of routing information is not required, whereas the data regarding voter and vote casting must be kept confidential. Similarly, the messages exchanged with the server should be communicated confidentially. We claim that secure communication required in e-voting cannot be guaranteed with a single keying scheme. Hence, a multiple key establishment for each PS is proposed in EVAS. The keys and their functions are described below.

1. GK

It is a shared key used by the server to communicate with the PSs. Every PS knows the GK in advance or it has built-in keys in the PS. When the server wants to communicate like transmitting the IKs to the PS, it will be encrypted with the GK. The PSs at the receiving end will decrypt the data with this global public key.

2. IK

The PS will use this key to encrypt the vote data while transmitting to the server. The server at the receiving end will decrypt the voting data for vote accounting purposes.

3. PK

When two PSs want to communicate with each other, the PK will be used. The receiver PS will decrypt the

Algorithm 6: Pseudo code of EVAS**Assumptions**

All the polling stations know the global key in advance.

1: **procedure** evoting-EVAS

2: **Input:** GK, voterdata

3: **Output:** result

// The server encrypts IK for each polling station using GK

4: for (i=1; i<=n; i++) do

5: Encrypt IK_i using GK

6: Broadcast IKs

7: end for

//Each polling station decrypts the received data to obtain their IK using GK

8: while (n!=null) do

9: Decrypt received data using GK

10: Obtain IK

11: end while

// Each polling station communicates PKs to its neighbors station, or the PSs directly connected to it.

12: for (i=1; i<=n; i++) do

13: Transmit PK_i to neighbor

14: end for

// The PS encrypts the voting data using IK when want to transmit. The data are sent to the server using multi-hop communication.

15: ENCRYPT (VOTE, IK)

16: while (receiver node !=server) do

//The encrypted data are again encrypted with the PK by the PS when a direct contact with the server is not possible.

17: ENCRYPT (VOTE, PK)

//The encrypted data are then transmitted to the neighbor PS.

18: Transmit to neighbor PS

//The receiver PS decrypts the received data with its PK.

19: DECRYPT (VOTE, PK)

20: end while

//The server then stores the data in the database in encrypted form.

21: STORE VOTE

22: if (election == end) then

//The data are retrieved from the database one by one,

23: RETRIEVE DATA

// decrypted and

24: DECRYPT (VOTE, IK)

// counted for every candidate in the contest.

25: COUNT (VOTE)

26: end if

27: **end procedure**

data before transmitting to the server. The PK will be known to the PSs that want to communicate. The process of this security mechanism is shown in Algorithm 6.

The establishment and update schemes of GKs, PKs, and IKs for each PS by EVAS are described in the following subsections.

Individual PS key establishments. The key that each PS share with only server is IK. Before deployment of each PS, the IK must be generated and pre-loaded into it. Each PS has a unique ID and the establishment of IK K map for a PS p is carried out as follows

$$K_{map} = fK_{ma}(p) \quad (1)$$

where f represents the pseudo random function and K_{ma} represents the master key. The master key is only shared with the server. When a server wants to communicate the PS p , it calculates the master key for the specific PS. The pseudo random function is used to reduce the computational overhead while computing the IK.

PK establishment. The focus of this work is the establishment of PKs known only to the PSs and their direct neighbors. A PS continuously exchange messages with its instant neighbors. For PSs whose neighborhood association is known in advance, for example, location of PSs before deployment, PK formation can be carried out easily by populating the PSs with their concerned PKs in advance. Our proposed system also assumes the deployment of new PSs in an e-voting system that are unaware of their neighbors. We believe that a PS installed in a security risk location must be designed in such a way that reduces possible attacks to a minimum for at least a small period (may be a few seconds) when caught by an opponent; else, the opponent might simply compromise all the PSs in an e-voting system and then control the system.

Key pre-distribution. The initial key K_I is generated and loaded to the PS by the controller. Each PS p originates a master key

$$K_p = fK_I(p) \quad (2)$$

Neighbor discovery. After deployment, PS p first sets a timer T_{min} and broadcasts a HELLO message. The message comprises PS id and some other useful information. PS then waits for the acknowledgment message that a neighbor node q respond. The acknowledge message (ACK) contains the identity of the PS q . The authentication of the ACK message received from PS q can be measured with master key K_q and is derived as $K_q = fK_I(q)$. As PS knows K_I , it can originate K_q and then authenticate PSs q identity

$$P \rightarrow * : p.q \rightarrow p : q, MAC(K_q, p|q) \quad (3)$$

Establishing PK. PS p calculates its PK K_{pq} with q as $K_{pq} = fK_p(p)$. PS q can also calculate K_{pq} similar to PS p . In this case, the PK will be K_{pq} . In this phase, no information need to interchange between p and q . No special communication is required for PS p , to authenticate itself with PS q , as any upcoming communications authenticated with K_{pq} by PS p will verify PS p 's identity.

Key erasure. It is not possible in EVAS that an untrusted server may operate because it does not have access to the GK. If it find GK some way, then it cannot communicate with the PS because the master key will be required to decrypt data coming from a PS. The PS p destroys all master keys K_q 's and K_I when its timer expires. Its own master key will remain its master key in the future and does not be removed. All PSs retains its particular master key.

When the above steps are completed successfully, PS p has established a PK with all its neighbors. Now, the PK will be used to encrypt the data that two neighbors want to communicate with each other. It is not mandatory for two PSs to use one key for sending data from one end and a new key in the opposite direction throughout their secure communication. Moreover, no PSs in the e-voting system own K_I . An opponent may have snooped on the entire traffic flow in this period, but without K_I it cannot vaccinate specious data or decipher any of the packets. An opponent is not able to compute the keying information of other PSs even if the attacker knows the keying information of a PS. When a compromised PS is noticed, the nearby PSs just erase the keys that were exchanged with it.

Establishing GKs. A key that is shared with all PSs in the e-voting system is the GK. This key is basically used when a server wants to share a confidential information with all the PSs in the system. The information may be a query on some event of concern. Hop by hop translation can be used if the server wants to distribute a message M securely to all PSs. More precisely, the message M is encrypted with its GK by the server. On receiving end, each PS that received the encrypted message needs to decrypt the message with GK in order to obtain the original message, M . The message is re-encrypted with its own key by each PS before sending it to its neighbors. The neighbor decrypts the message received with its neighbor key to obtain the message M . The message is re-encrypted by each neighbor before sending it to its neighbors. This practice is repeated till all the PSs receive M . However, the key limitation of this scheme is the computation overhead of each PS to decrypt the message in order to obtain the original message and re-encrypt before sending it to its neighbors. Thus, the most preferable way is the use of GK for broadcast message encryption.

Loading GK to every PS before deployment is the easiest way to bootstrap a GK for e-voting network. A key concern that arises instantly is the necessity to securely change, GK once a compromised PS is noticed. In other words, the GK must be updated and broadcasted to all the outstanding PSs in a reliable, secure, and timely manner known as global re-keying.

The communication complexity of key distribution to individual PSs by the server via the global re-keying based on unicast is $O(N)$ keys. Here, N is the number of PSs in e-voting network. To reduce the complexity of re-keying operation, logical key trees may be used. It is to be noted that the entire re-keying message is sent to all PSs during key distribution, but the polling needs a small fraction of this message. Physical locations can be used to map the PSs to logical key tree in order to decrease the unnecessary data. But, this may result in large overhead. Furthermore, the key server needs full information about the topology of the system in order to work fine on this scheme. Suppose, we have $N = 1024$ PSs. In case of binary key tree, the number of keys that need to be broadcasted is 10. If the size of each key is 10 bytes and a packet comprised 29 bytes. To avoid fragmentation of keys, the key server needs five packets to be broadcasted. The distribution of packets is carried out in a hop by hop manner to all PSs in a reliable way.

Local broadcast authentication. The authentication of every message before forwarding is one of the compulsory constraints for a secure e-voting system. Two broadcast authentication schemes are used, for example, global broadcast and unicast authentication. In global broadcast authentication, the server authenticates a packet to all PSs, while in unicast authentication, the PS is responsible for authenticating a packet before forwarding to their neighbors. Authentication of local broadcast is mandatory to support passive participation. It is to be noted that locally broadcast messages are typically time or event driven, such as the periodic broadcast of control information from a PS. The next packet that a PS transmits is not known in advance.

One-way key chain based authentication. The use of one hop broadcast authentication known as one-way key chain²⁷ is used in this proposal. Delay disclosure and time synchronization among neighboring PSs are not required in this scheme. Initially, one-way key chain of definite length is generated by each PS, the keychain is then encrypted with the PK before transmission to their neighbors. The key used for authentication in a PS's one-way key chain is referred to as authentication key (AK) in this scheme. The next AK in key chain is attached to the message whenever a PS wants to communicate with their neighbors. The reverse order is used to disclose the AK at the receiver end. The message authentication is performed on the receiver PS using AK received before. Designing our authentication mechanism is driven by two observations. First, authentication of packets (control information) transmitted to immediate neighbors of the PS is required. Second, when the PS transmits a packet, the neighbor

usually receives the packet prior any other PS receives a transmitted copy. This is true due to the difference of PS distances involved in the system. Once PS p transmits a packet that holds the message M and an AK , before receiving a forwarded copy from PS x , PS q already received the packet because $|pq| < |px| + |xq|$. So, when PS v gets the message M , the opponent x cannot use again the AK key K to vaccinate additional packet during impersonation $|pq| < |px| + |xq|$. Source authentication and passive participation can easily be provided with the above scheme.

Security analysis

The security of the keying schemes in EVAS is analyzed in this section. The survivability of the e-voting system when undetected compromise occurs is discussed. Then, the robustness of defense mechanism of our proposed schemes against different types of attacks is elaborated.

Survivability

When a PS p is compromised, the opponent can use PS p 's keying information to initiate attacks. Our scheme has the potential to detach the compromised PS quickly from the system using global re-keying mechanism if a compromise event is detected. The opponent cannot initiate more attacks after the reversal. Hence, we consider that survivability under conditions where PS compromises are not yet detected is one of the most serious security needs of the e-voting systems. The rest of the section is dedicated to the general consideration of what an opponent can do when compromising a PS. The details on attacks are discussed in section "Defending against various attacks on secure routing."

Stealing the IK lets the compromised PS to vaccinate incorrect data reading, when PSs send their readings straight to the server authenticated with IK. Next, an opponent can build trust with neighboring PSs when the PKs of a compromise PS are obtained. Hence, the opponent can insert some malicious information to control messages in the system. However, because of our one time key based authentication approach, the identity of the compromised PS can be used to initiate such attacks. The important feature of the proposed scheme is its capability of limiting the probable damage, since each station maintains a list of trusted neighboring after the e-voting stations placement. Therefore, trust relationship cannot be established with the compromised PSs other than their neighbors and the secure links between PSs cannot be placed at risk.

The opponent can decrypt the broadcast messages by the server if it obtains the GK. For example, GK can be extracted easily if the opponent compromises a single station. The broadcast messages are planned to be known by every station, so the information can be

revealed easily whatever scheme for secure message distribution is used. Furthermore, knowing the GK does not empower the opponent to flood the whole system with malicious data while impersonating the server, since any information sent by the server is authentic. To conclude, the opponent can only decrypt the messages that are encrypted with the current key. In our proposal, periodic global re-keying mechanism is deployed and after some period, the opponent will be unable to decrypt the messages.

Defending against various attacks on secure routing

The outsider attacks are prevented in EVAS using local broadcast authentication in which the control messages used for routing are authenticated. The wormhole attack cannot be prevented via local broadcast authentication scheme and its prevention is discussed in section "Dealing with the wormhole and sinkhole attacks." Hence, in this conversation, we generally consider attacks initiated by an opponent inside the e-voting system that has compromised one or more PSs. The type of attacks that an opponent may launch inside the network can be spoofing, repelling, or attracting network traffic, false error message generation, and replay or alter routing information in the hope of forming routing loops. The opponent may also initiate the selective forwarding attack. In this attack, the compromised PS destroys the packets regarding routing information and forwards other packet in a reliable way. Initiation of these attacks cannot be prevented, but the consequences of these attacks can be minimized by our proposed scheme. First, the attacks are restricted to two hop zone of the compromised PS due to local broadcast authentication. The detection of these attacks is not very difficult and the opponent will take a high risk to launch it as these attacks are localized in a small zone. Second, the updating attack is also highly likely to be identified since the sending station may eavesdrop its packet being changed while being forwarded by the PS that is compromised. Third, the compromised PS can be revoked from the system very efficiently using global re-keying once detected. The following attacks can be prevented by our proposed scheme. The Hello flood attack is launched in such a way that the message is broadcasted with high power that may convince each PS about their neighbor. Upon successful attempt of hello flood attack, all the PSs send their readings into oblivion. Conversely, this attack will not succeed in EVAS since packets will be accepted from authentic neighbors only and thus can prevent the Sybil attack.²⁸

Dealing with the wormhole and sinkhole attacks. The prevention and detection of the wormhole and sinkhole

attacks launched at once is one of the most difficult tasks in security assurance. Sinkhole attack²⁹ is an attempt of a compromised PS to attract packets, for example, voting data, by presenting information such as high end-to-end reliability, from its neighbors and then drop them. The validation of this information is a difficult task. In the wormhole attack,³⁰ normally two malicious PSs that are away from each other are linked with an invisible link to the fundamental e-voting system having low latency. These PSs fake a route that is shorter than original one. This attack confuses routing scheme that is based on PS distances. When deploying one such PS near to the server and the other near to the attacking station, the opponent could assure the PSs close the target that they are only one or two hops distance from the server. However, this would usually be several hops distance from the server. Therefore, it produces a sinkhole. In a similar way, in wormhole attack, PSs that are at the distance of multiple hops may consider that they are neighbors. It is a very strong attack, since the opponent does not have to compromise any PS to be able to initiate it. In EVAS, an external opponent cannot successfully launch a wormhole attack at any time other than a neighbor discovery. Neighbor discovery can be used for the processing of the PK. All the neighbors of a PS are identified after this stage. Therefore, the opponent cannot convince the two remote stations that they are neighbors. The probability that the opponent succeeds in such an attack will also be very small, because the time for neighbor discovery is very low (in the order of a few seconds) relative to the election time. We note that knowledge of the authenticated neighborhood knowledge is essential for defending wormhole attacks. In EVAS, an internal opponent must compromise at least two PSs to create a wormhole. Despite that, it still cannot assure the neighborhood of two remote stations, even after the process of finding the location of neighbors is completed.

However, if an attacker compromises a PS p nearby server, Q is in another area of interest, it can successfully create the PS q as a sinkhole because the number of stations between p and the server becomes smaller, making PS q particularly attractive to nearby stations. In our scheme, the location of the server known in advance, so every PS knows the number of hops between the server and the PS after establishing network topology. As a result, it is difficult to create a very attractive sinkhole for the opponent without being noticed.

Experimental evaluation

The dependent variable efficiency, effectiveness, and satisfaction are used to evaluate the performance of our proposed e-voting scheme in this section.

Effectiveness: this metric was examined by evaluating the errors in ballot tabulated by the contest, on the basis of ballot and based on error type. The deviations from the slate represent all the errors

Efficiency: the ballot completion time was recorded to measure the efficiency of the proposed voting system against manual system. The information was gathered using a stop watch when the voter enters the PS. The stopwatch was started on the entry and stopped when a vote leaves the polling booth.

Satisfaction: this metric was evaluated through a questionnaire in which 16 questions were asked from the voter after casting their vote in addition to age and education. The survey was conducted after the voter cast their vote to know about the immediate impression of the voters.

Materials

One e- voting system and a paper-based voting system were used in this experiment. First, the proposed e-voting system was a customized software programmed totally in Java2e. The easy and usable interface was designed in such a way that is capable of providing a voting experience in a simulated election environment which mimics the real life election. The voter's authentication is achieved through a biometric verification system. After biometric verification, the voter information is displayed automatically. The instructions on the screen easily guide the voter, where the voter is capable to change the choice once made before voter submission. The candidates are selected by simply touching the radio button in front of each candidate name and symbol. After selecting the candidate, a review screen was presented with their voting choices made, which helps to confirm the right candidate selection. After viewing the review screen, the voter is able to submit their votes. Finally, the voter is asked to fill a questionnaire that records the age, education, and other usability factors.

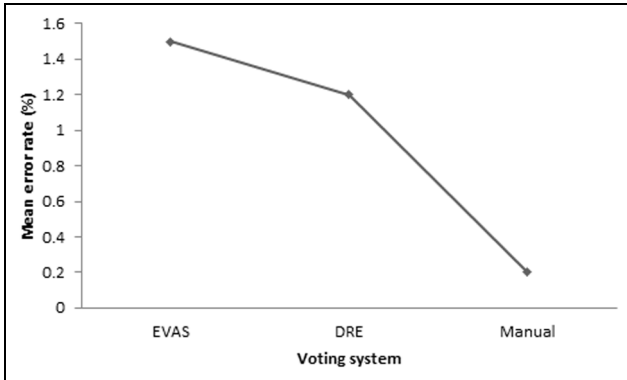
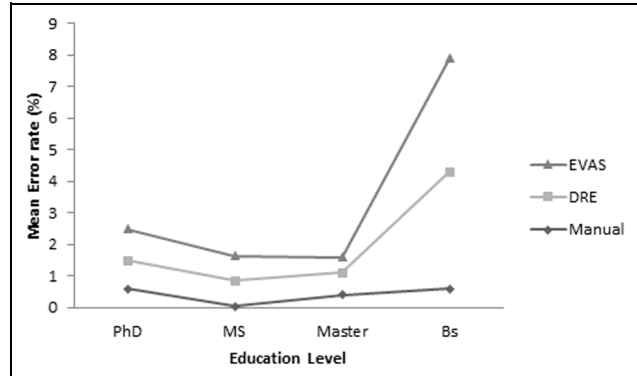
Results

The voters, whose age is below 20 and made more than five errors on both manual and e-voting systems, are considered outliers. The data associated with outliers are not included in the analysis. Some of participant voters declined to vote according to the schedule provided to them. Political ideologies are the base for voting and declined to vote as shown from the verbal comments of the voter. One other voter was discarded from the comparative analysis due to a technical error that disallowed recording the data from e-voting system. Likewise, three diverse voters were discarded from the evaluation of ballot completion times. The voter having minimum one ballot completion time is

Table 2. Distribution of errors per voting system.

Voting system	Ballots cast	Total errors	Ballots with at least one error	Mean errors	SD
Proposed e-voting	100	88	13	0.64	2.57
Manual voting	100	31	9	0.22	0.97

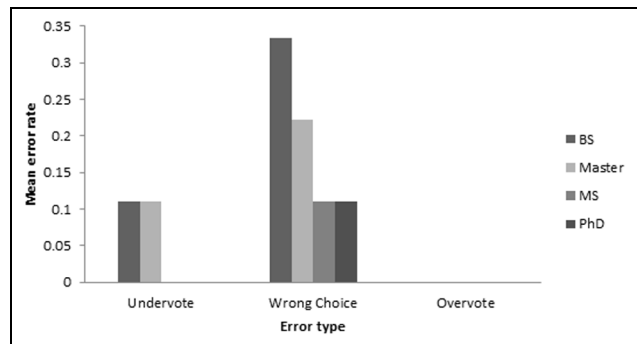
SD: standard deviation.

**Figure 1.** Mean error rate (%) as a function of voting system.**Figure 2.** Mean error rate (%) as a function of voters self-reported level of education and voting system.

exceeded three inter-quartile ranges. The voting error rates were analyzed using co-variance. In the first analysis, two variable voting systems, error type, education were used for experiments. The age is considered as a co-variant and is applied to the factors that are common in both voting systems, for example, voter education. Since the age is not a reliable predictor, it is not used in further analysis. However, surprisingly the e-voting systems produce more errors than manual voting systems (as shown in Table 2). The voting errors are one-third on a manual system while the proposed e-voting scheme produces more errors.

While comparing the results as shown in Figure 1, transversely all other issues, there was a key concern of voting system such that the e-voting system produced the maximum error rates from voters. Voters' education level moderated this relationship among voting system and error rate (as shown in Figure 2) such that the level of education of voter influences the results. The low level educated voters are more committed in errors as compared to voters having higher education. Education level of voters also had an influence on the error's type that they made. The voters having a low education level are more prone to errors regarding choices while casting the votes both in electronic and manual voting systems, as shown in Figure 2.

In Figure 3, the results of wrong choices during an election are shown in both voting systems. In this experiment, the voting systems, error type, and education of voters are considered. The age is used as a co-variant. Other factors like computer proficiency,

**Figure 3.** Mean error rate as a function of education level and type of error (proposed e-voting).

display method that is applicable to e-voting systems are considered. As shown in Figure 4, both e-voting systems are more prone to errors made in the election process.

Efficiency. Co-variance was used to analyze the error rates in previous experiments. In these settings, the analysis was conducted on the basis of education, slate candidate's position, and voting system. Here, the age of the voters was considered as a co-variance. All other factors that are applicable to the voting systems under consideration were also incorporated.

Furthermore, the age of voters (co-variant) was a reliable parameter statistically and was considered in

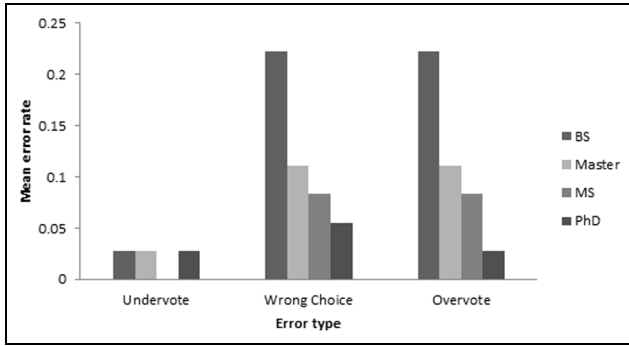


Figure 4. Mean error rate as a function of education level and type of error (manual).

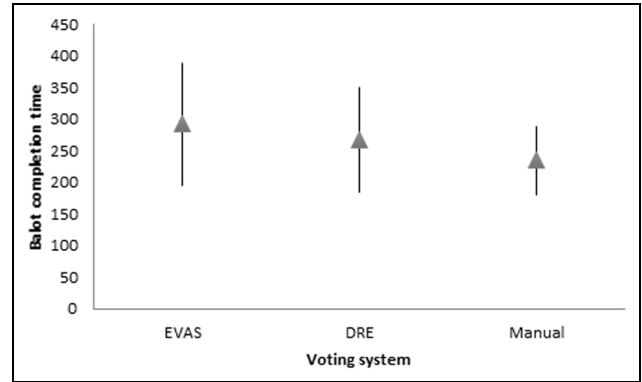


Figure 6. Ballot completion time of each voting system.

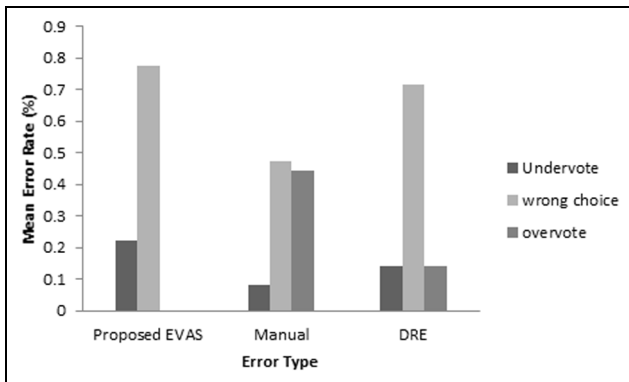


Figure 5. Mean error rate as a function of error type and e-voting system.

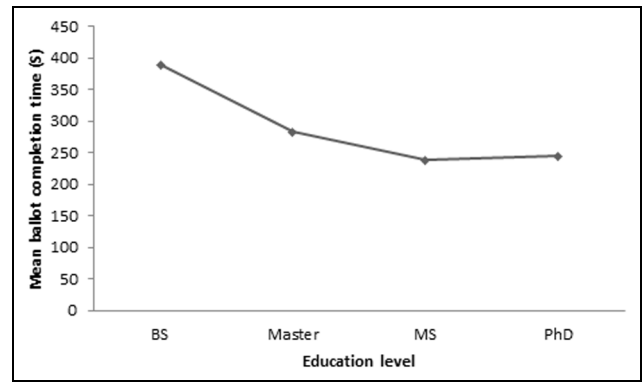


Figure 7. Mean ballot completion time as a function of voter education level.

the experiment. It shows that the elder voters took longer to complete the ballots. The slate candidate’s position has no visible influence on the results, so it is not considered in the experiments further. Electors took somewhat slower to complete their ballots while casting on the proposed e-voting system as compared to manual voting as shown in Figure 5. The manual system was the slowest of both, though, this outcome is highly likely to be insignificant as the degree of the result was fairly small, about 60 s among the paper ballot and the proposed e-voting system. In addition, the overall time required for manual vote casting is quite large. As finding eligible voters in the voter list and then marking each voter is a cumbersome task and may introduce additional delays. Hence, the time required for operating the proposed voting system may be compensated because no authentication is required before entering to the polling booth. The voter’s level of education also influences the results of ballots completion time. The voter education was reported by the voters. However, the fundamental correlation is not completely clear, across voting systems and ages. As shown in Figure 6, the self-reported low education level of the voter population took longer while completing their ballots.

Subjective satisfaction. In this section, e-voting system of usability scale ratings is used to analyze the usability of the system. Similar to other experiments, the parameters in this analysis are voting system, slate candidate’s position, voter education with age as a co-variant. However, the level of education and slate position have no visible influence on the results, so it is neglected in later discussions. The allocations of system usability scale ratings were somewhat negatively skewed for all voting systems. However, more skewedness can be seen in the proposed EVAS. This is probably because usability scale ratings are sinking at or close to the maximum score of 100. Moreover, the age factor (co-variant) was statistically consistent, responsible for 8% of the difference in system usability ratings in all voting systems, demonstrating that mature voters inclined to be the most precarious raters in all voting systems under consideration. To close, fully reliable with earlier research, the voting systems under consideration received satisfactory usability scale scores for all ages. As shown in Figure 7, the DRE voting system was rated at the maximum, the proposed e-voting system at the second highest, and the manual voting system at a similar performance to the proposed system (Figure 8).

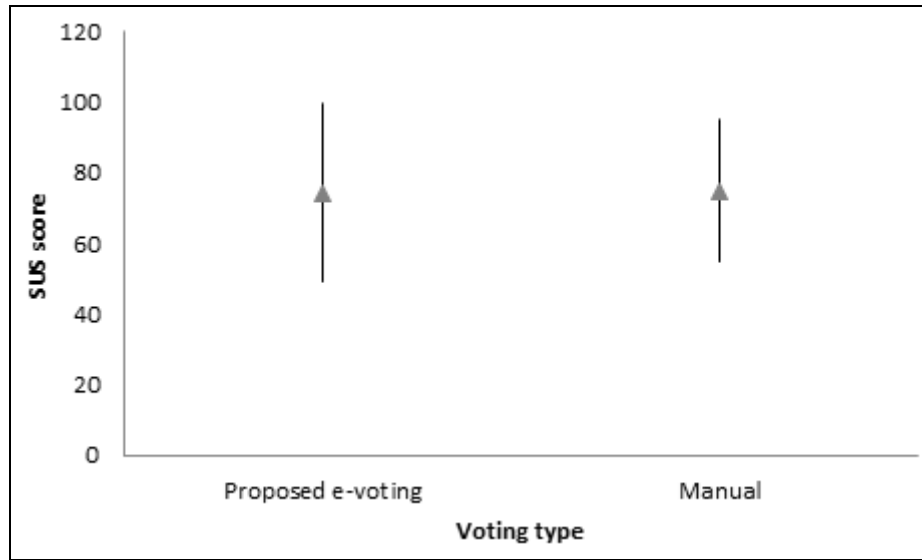


Figure 8. System usability scale scores as a function of voting system.

Discussion. The outcomes from first experimentation noticeably show three indications. First, the voters with low education are typically vulnerable to make errors in e-voting systems. However, it is not clear whether this was a result of lower understanding of computer use in this test or other mitigating issues. Subjective proof from the voter's observation recommends that it is probably attributable to the absence of familiarity or information about how to operate the proposed e-voting system's user interface. It is a matter of discussion, however, the results obtained in the virtual election conducted for the purpose of the experiment have influenced the results as the election environment was artificial in nature. It is a matter of discussion, however, the results obtained in the virtual election conducted for the purpose of the experiment have influenced the results as the election environment was artificial in nature. The list of contesting candidates were given to the voters before casting their votes. Some voters were trained before casting their votes. The error ratio in less educated voters may be high because of their uncommitted behaviors toward the virtual election system. Another factor is the review screen. In the proposed system, the review screen was on a single page. The candidate names are displayed on a single screen with small font size. In this environment, the importance of the information displayed on the screen and its presentation cannot be ignored when a voter is making choices. In this context, the review screen is another form of a long ballot in which the information is displayed in a different way ranging from making selection to verification of choices.

Conclusion

In this article, the usability, security, and authentication of the proposed e-voting system is discussed. A

new voting system is developed using Java. The voter is first authenticated by the biometric verification system. The eligible voters can cast their votes in a few easy steps. After the voter cast their vote, the voter data are communicated to the server in a secure way. Our e-voting solution EVAS is proposed as a key management scheme. Security of PS in e-voting systems cannot be guaranteed with single key encryption schemes. The attacker can reveal all the information about the network when a single PS is compromised. In order to guarantee authentication of different types of messages communicated in the network, multiple key concept is proposed. EVAS supports three different types of keys per PS in order provide authentication. The GK allows the server to communicate with all PSs. The PK is utilized for secure transmission between PSs. The IK is used for sending secure messages from PSs to the server, which will only be shared with server. These keys can be used to enhance the security of numerous security schemes. EVAS contains a well-organized procedure using one-way key chains for local broadcast authentication. EVAS can restrict or enhance the effort of initiating numerous security attacks on e-voting systems. The establishment of keys and their updates can be performed very efficiently both in terms of computation and storage requirements. An experiment for the proposed e-voting system was conducted at the University of Malakand, where the students took part in an election set up. The usability of manual voting and e-voting was evaluated. The students were asked to fill a questionnaire (as provided in Appendix 1) after casting their vote. The students' feedback and voting behavior were recorded. The data are then analyzed and their results are discussed. The results demonstrated that the proposed e-voting model is practical

and secure as compared to other e-voting systems proposed in the literature.



Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iDs

Ateeq Ur Rehman  <https://orcid.org/0000-0001-5721-0867>
Muazzam A Khan  <https://orcid.org/0000-0002-2713-5605>

References

- Zissis D and Lekkas D. Securing e-government and e-voting with an open cloud computing architecture. *Govern Inform Quarter* 2011; 28(2): 239–251.
- Khan NU and Akhter S. Historical challenges to Pakistan's good governance: reforms in the election process in conducting free, fair and transparent elections. *History Pakistan Stud Punjab Univ* 2016; 30(2): 152–167.
- Weerakkody V, Irani Z, Lee H, et al. e-government implementation: a bird's eye view of issues relating to costs, opportunities, benefits and risks. *Inform Syst Front* 2017; 17(4): 889–915.
- http://democracy-reporting.org/wp-content/uploads/2016/02/dri_kuwait_report_08.pdf
- Berghel H. Digital politics 2016. *IEEE Comput* 2016; 49(1): 75–79.
- Gibson JP, Krimmer R, Teague V, et al. A review of e-voting: the past, present and future. *Ann Telecommun* 2016; 71(7): 279–286.
- Ahmed TK and Aborizka M. Secure biometric e-voting scheme. In: *International conference on intelligent computing and information science*, Chongqing, China, 8–9 January 2011 (ed. R Chen), pp. 380–388. Berlin: Springer.
- Bernardi L, Sandri G and Seddone A. Challenges of political participation and intra-party democracy: bitter-sweet symphony from party membership and primary elections in Italy. *Acta Politica* 2017; 52: 218–240.
- Gallegos-Garcia G and Tapia-Recillas H. Electronic voting protocol using identity-based cryptography. *Sci World J* 2015; 2015: 741031.
- Pigatto DF, Gonsalves L, Roberto GF, et al. The HAMSTER data communication architecture for unmanned aerial, ground and aquatic systems. *J Intel Robot Syst* 2016; 84(1): 705–723.
- Tretyakov DB, Kolyako AV, Pleshkov S, et al. Quantum key distribution in single-photon communication system. *Optoelectron Instrum Data Pr* 2016; 52(5): 453–461.
- Karayumak F, Kauer M and Olembo MM. Poster: usable verifiable remote electronic voting—usability analysis of the Helios system, 2011, https://cups.cs.cmu.edu/soups/2011/posters/soups_posters-Karayumak.pdf
- Al-Ibrahim M and Al-Ostad J. *The usability, credibility and security of e-voting system in education sector*, <http://www.ipedr.com/vol27/21-IC4E%202012-F00028.pdf>
- Karayumak F, Olembo MM, Kauer M, et al. Usability analysis of Helios—an open source verifiable remote electronic voting system, vol. 11, https://www.usenix.org/legacy/event/evtwote11/tech/final_files/Karayumak.pdf
- Mac Namara D, Gibson P and Oakley K. Just Like Paper—a baseline for classifying e-voting usability, p.113, <https://www.semanticscholar.org/paper/Just-Like-Paper-a-baseline-for-classifying-e-voting-MacNamara-Gibson/5831c8bd356bdb74622c516c09d7406b216863e6>
- Uzunay Y and Bicakci K. Trusted3Ballot: improving security and usability of three ballot voting system using trusted computing. In: *5th international conference on intelligent systems, modelling and simulation*, Langkawi, Malaysia, 27–29 January 2014, pp.534–539. New York: IEEE.
- Bruns D, Do HQ, Greiner S, et al. Poster: security in e-voting, https://www.ieee-security.org/TC/SP2015/posters/paper_10.pdf
- Campbell BA, Tossell CC, Byrne MD, et al. Voting on a smartphone: evaluating the usability of an optimized voting system for handheld mobile devices. *Proc Human Fact Ergon Soc* 2011; 55: 1100–1104.
- Khairnar S and Kharat R. Survey on secure online voting system. *Int J Comput Appl* 2016; 134(13): 19–21.
- Walake MA and Chavan MP. Efficient voting system with (2, 2) secret sharing based authentication. *Int J Comput Sci Inform Tech* 2015; 6(1): 410–412.
- Al-Anie HK, Alia MA and Hnaif AA. e-voting protocol based on public key cryptography. *Int J Netw Secur Appl* 2011; 3(4): 87–98.
- Binu V, Nair DG and Sreekumar A. Secret sharing homomorphism and secure e-voting, 2016, <https://arxiv.org/abs/1602.05372>
- Desmedt Y and Erotokritou S. Towards usable and secure internet voting, https://www.researchgate.net/publication/235673057_Towards_Usable_and_Secure_Internet_Voting
- Zwierko A and Kotulski Z. A light-weight e-voting system with distributed trust. *Electron Note Theor Comput Sci* 2007; 168: 109–126.
- Kim MS, Park SK, Kim HS, et al. An efficient key management scheme for advanced metering infrastructure. In: DS Park, HC Chao and YS Jeong (eds) *Advances in computer science and ubiquitous computing*, pp.125–130. Singapore: Springer.
- Falkner S, Kieseberg P, Simos DE, et al. e-voting authentication with QR-codes: human aspects of information security, privacy, and trust. In: *Proceedings of Second International Conference, HAS 2014* (eds T Tryfonas and I Askoxylakis), June 2014, pp.149–159. Berlin, Heidelberg: Springer-Verlag.
- Kar N, Roy S, Saha A, et al. A biometric based design pattern for implementation of a security conscious e-voting system using cryptographic protocols. In: *Mobile communication and power engineering: second international joint conference, AIM/CCPE*, Bangalore, India, 27–28 April 2012, pp.78–85. Berlin: Springer.

28. Sood M and Vasudeva A. Perspectives of Sybil attack in routing protocols of mobile ad hoc network. In: *Computer networks and communications (NetCom): Proceedings of the fourth international conference on networks and communications*, February 2013, pp.3–13. Berlin: Springer.

29. Krontiris I, Dimitriou T, Giannetos T, et al. Intrusion detection of sinkhole attacks in wireless sensor networks. In: *Algorithmic aspects of wireless sensor networks: third international workshop, ALGOSENSORS*, Wroclaw, 14 July 2007, pp.150–161. Berlin: Springer.

30. Upadhyay S and Chaurasia BK. Detecting and avoiding wormhole attack in MANET using statistical analysis approach. In: *Advances in computer science and information technology. networks and communications second international conference, CCSIT*, Bangalore, India, 2–4 January 2012. Berlin: Springer.

Appendix I

Questionnaire

Annex A							
Questionnaire							
E-Voting Usability Questionnaire							
Age: _____				Class: _____			
Please tick your choice							
	SCREEN		1	2	3	4	5
1	Reading characters on the screen	Hard					Easy
2	Highlighting simplifies the task	Not at all					Very much
3	Organization of information	Confusing					Very clear
4	Sequence of screens	Confusing					Very clear
	TERMINOLOGY AND SYSTEM INFORMATION		1	2	3	4	5
5	Use of terms throughout system	Inconsistent					Consistent
6	Terminology related to the task	Never					Always
7	Position of messages on screen	Inconsistent					Consistent
8	Prompts for input	Confusing					Clear
	LEARNING		1	2	3	4	5
9	Learning to operate the system	Difficult					Easy
10	Remembering names and use of commands	Difficult					Easy
	Performing tasks are straightforward	Never					Always
11	Help messages on the screen	Unhelpful					Helpful
	SYSTEM CAPABILITIES		1	2	3	4	5
12	System speed	Too slow					Fast enough
13	System reliability	Unreliable					Reliable
14	The system tends to be	Noisy					Quiet
15	Correcting your mistakes	Difficult					Easy
16	Designed for all levels of users	Never					Always

Figure 9. E-voting usability questionnaire.