# An Electronic Voting Protocol with Deniable Authentication for Mobile Ad Hoc Networks *

Chun-Ta Li‡ Min-Shiang Hwang† Chi-Yu Liu§

Department of Management Information Systems†
National Chung Hsing University
250 Kuo Kuang Road, Taichung, Taiwan 402, R.O.C.
Email: mshwang@nchu.edu.tw
Fax: 886-4-23742337

Department of Computer Science and Engineering‡
National Chung Hsing University
250 Kuo Kuang Road, Taichung, Taiwan 402, R.O.C.

Graduate Institute of Networking and Communication Engineering§
Chaoyang University of Technology
168 Gifeng E. Rd., Wufeng Taichung County, Taiwan, R.O.C.

September 14, 2012

*This work was supported in part by Taiwan Information Security Center (TWISC), National Science Council under the grants NSC 95-2218-E-001-001, and NSC95-2218-E-011-015.

†Responsible for correspondence: Prof. Min-Shiang Hwang

# An Electronic Voting Protocol with Deniable Authentication for Mobile Ad Hoc Networks

**Abstract**

In this article, we propose a deniable electronic voting authentication protocol for mobile ad hoc networks, which meets the essential requirements of a secure e-voting system. Due to the characteristics, constraints, and security requirements of mobile ad hoc networks, our protocol does not require the aid of any centralized administration and mobile nodes could cooperate with each other to securely facilitate e-voting. Finally, the proposed protocol provides the ability to deniable authentication. When a legal voter casts a vote with this voting system, he/she can deny that he/she has voted for a third party and the third party cannot judge who votes.

*Keywords:* Cryptography, e-voting, deniable authentication, mobile ad hoc networks, security.

## 1 Introduction

Supporting group decisions has become an important topic in the field of computer applications, and electronic voting (e-voting) has received a great deal of attention in recent decades. People can use modern digital devices such as PCs, PDAs, cell phones, or laptops and networks such as Internet, Intranet, wireless networks, or ad hoc networks to make group decisions electronically. Let us examine the following scenario.

Jack is a general manager and he comes to inspect a new factory with the board of directors. Then, his secretary tells him and directors that there is a pressing emergency and that it immediately needs them to be decided by vote. Moreover, it is to be a silent vote and their mobile devices such as PDAs are unable to access the Internet to reach the online e-voting system. Is there a simple and secure way for them to use their own mobile devices to place their votes? An ad hoc mode e-voting seems applicable to this scenario. Just as existing e-voting systems attempt to do, our e-voting protocol for mobile ad hoc networks is designed to fulfill the requirements shown below.

## 1.1 General Requirements of E-Voting

In this subsection, we will briefly describe the essential criteria that a secure e-voting system should satisfy the following requirements:

**Completeness:** A voting system would be called completeness if it is unable to fake a vote, unable to remove a valid vote from the final tally, and unable to add an invalid vote to the final tally.

**Uniqueness (Unreusability):** Each of eligible voters can cast their vote only once. In other words, the voting system must prevent double voting.

**Privacy:** Although an intruder can eavesdrop on the messages that are transmitted between a voter and the electoral unit (the voting center), the transmitted vote itself is anonymous. In other words, no one can link the ballot to the voter.

**Eligibility:** Before beginning the vote, each voter must pass a series of voter authentication processes. When the voter passes the authentication process, he/she is permitted to join the vote. In other words, only eligible voters are allowed to vote.

**Fairness:** During the voting phase, no one can know the partial results of the election and all the voted ballots must be kept secret until the end of the voting session. In other words, no transitory information is learned before the tally result is published.

**Verifiability:** For a voting system, anyone should be able to independently verify that all legitimate votes have been counted correctly. In this case, the system has achieved strong verifiability. Weak verifiability means that the voter only can verify his/her own vote.

**Uncoercibility:** The voter must be able to cast the vote according to his/her own conscience. In other words, no voter can be force to vote in a particular way thanks to the prevention of bribery and extortion.

**Mobility:** A voting system would have achieved mobility if there is no restriction on the designated location in which voter can cast his/her ballot.

**Efficiency:** The computational loads must be light and able to be performed within a reasonable amount of time. In other words, no heavy computational load through the duration of the election.

**Scalability:** Due to the fact that messages exchanged for the vote consume significant CPU cycles and wireless bandwidth, an e-voting system has to take the communication rounds between voter and voting system into account during the whole course of the election.

**Deniable authentication:** When a voter casts a ballot to the voting center, the ballot should been verified only by the voting center and the voting center cannot prove the ballot to any third party, even if he/she fully-cooperates with the third party. In other words, the voter can deny his vote to a third party.

## 1.2 Ad Hoc Networks and Their Security Considerations

A mobile ad hoc network (MANET) can be quickly deployed as needed as it consists of some available mobile nodes with wireless network interfaces to form a temporary network. In MANETs, nodes use wireless radio technology to communicate with each other directly if they are both within wireless transmitter range. However, since there is no stationary infrastructure or centralized administration such as base stations, communication nodes must act as routers for themselves and also rely on other nodes to relay communication data. Moreover, several key features of MANETs must be taken into consideration if they are to be used [9, 20], including limited power, limited memory, and limited calculation capacity. In MANETs, security is a more important issue when compared to wired or other wireless systems [5, 24, 32, 33, 37, 38, 39].

A malicious node or an intruder can easily eavesdrop on the communication channels between ad hoc nodes and discovers sensitive information. This is known as, a passive or eavesdropping attack. Eavesdropping attacks can cause many threats to the security and privacy of the network. On the other hand, malicious nodes can inject false messages, alter them, or re-send them on the communication channels between nodes and frustrate the communication among these nodes in the network. This is known as, an active attack, and includes impersonation attacks, replay attacks, and man-in-the-middle attacks. As a result of the threat posed by the above-mentioned attacks, there are many cryptographic techniques commonly used to design security countermeasures for MANETs such as symmetric-key (secret key) [1], asymmetric-key (public and private key) [19], and one-way hash functions [25].

## 1.3 Our E-voting Protocol: Motivations and Goals

Usually, a mobile ad hoc network is deployed in a designated area without any fixed infrastructure or centralized authority. While most existing e-voting protocols are reliant upon a centralized and trusted third party (such as a registration center, monitor center, or publishing center), it is unfeasible to introduce them within an ad hoc network. Motivated by the basic differences between ad hoc and fixed networks and the characteristics of ad hoc networks mentioned above, we have re-examined and re-considered existing e-voting protocols and designed an applicable e-voting protocol for mobile ad hoc networks where mobile nodes would cooperate with each other to implement e-voting.

The basic idea of our e-voting protocol is that mobile nodes are organized into two roles to form an underlying service group for electronic voting. The first role is that of the chosen group leader [1] (also called the system), and the second is that of the other nodes that play the role of voters. To the best of our knowledge, this is the first attempt to an electronic voting protocol for mobile ad hoc networks with deniable authentication. Our e-voting protocol can not only achieve the above-mentioned essential requirements in Section 1.1, but they also can resist the above-mentioned passive and active attacks listed in Section 1.2, thereby increasing e-voting's security, and reliability, and can be practically applied to MANETs.

### 1.4 Outline

The remainder of this article is organized as follows. Section 2 reviews related works of e-voting and tools used in our protocol. Details of the proposed e-voting protocol are described in Section 3. A performance evaluation of the proposed approach and the comparisons between some related e-voting protocols and ours are conducted in Section 4. In Section 5, we conclude the article and discuss possible future work.

## 2 Related Works

In this section, we review some related works regarding the basic concepts of e-voting, and cryptographic techniques used in our protocol.

[1] Group leader election is a non-trivial issue in ad hoc networks and detailed technique on this can be found in [29, 36] and is out of the scope of this paper.

## 2.1 Previous and Related Works Regarding E-Voting

Chaum (1981) [7] proposed the first electronic voting method that enables voters to electronically cast his/her ballot over insecure networks. Later, there were persistent efforts made by researchers to increase security and to make the applications of e-voting more comprehensive. Several e-voting protocols have been proposed in the literature [4, 6, 8, 10, 21, 22, 26, 27, 31, 35].

Liaw (2004) [26] proposed an e-voting protocol using smart cards. In their protocol voters could use the receipt stored in the smart card to confirm whether his/her vote had been counted or not. If it has not been counted, the voter can ask the publishing center to recount his/her vote by sending the receipt. However, this approach lacks the requirement of uncoercibility [6] and thus it is difficult to prevent vote buying and extortion.

Chen et al. (2003) [8], and Chang et al. (2006) [6], introduced a public proxy server that has the advantage of increased anonymity during the voting phase. The voter sends his/her vote to the voting center through a trusted proxy server, and the original network address of voter can be replaced by a proxy address. Thus it is impossible to link a ballot to a voter or to trace the location of the voter's. However, both of their protocols are unable allow to voter to verify whether their vote has been counted or not and the requirement of verifiability has thus not be achieved.

Karro and Wang (1999) [22] proposed an online election protocol without using blind signatures for large scale elections. In their protocol, six servers are introduced to provide integrity and privacy of communication between the voters and the servers; and the link between the ballot and the voter is be broken thus guaranteeing the requirement of anonymity. However, practically speaking, their protocol seems too complex to implement [8].

To reduce the cost of management, Cranor and Cytron (1997) [10] proposed a well-known electronic voting protocol called Sensus that required only two servers, namely a validator and a tallier. However, Baiardi et al. (2005) [4] showed that Sensus suffers from the vulnerability that illegitimate votes would be counted in the final tally and the requirement of completeness would not be achieved. Also, they further proposed an improvement of Sensus called SEAS (Secure E-voting Applet System).

## 2.2 Hard Problem Assumptions

The security of our proposed e-voting protocol is under the assumption that the computational Diffie-Hellman (CDH) [12] problem is hard; that is, given $g$, $g^x \in Z_p^*$ and $g^y \in Z_p^*$, it is hard to compute $g^{xy}$, where $g$ is a generator of $Z_p^*$ and $p$ is a large prime.

Moreover, it is hard to invert an one-way hashing function $h(.)$; that is, given $x$ and $h(.)$, it is easy to compute $h(x) = y$. On the contrary, given $y$, it is hard to find $x$, satisfying $h(x) = y$.

Finally, it is hard to solve the discrete logarithmic problem [15]; that is, given a generator $g$ of $Z_p^*$ and $X \in Z_p^*$, it is computationally infeasible to find $x$, satisfying $X = g^x \bmod p$, where $p$ is a large prime.

## 2.3 Deniable Authentication

Dwork et al. (1998) [14], first proposed an application of zero-knowledge, deniable authentication protocols. Afterward, many paper followed his lead and furthered this work [2, 3, 16]. Deng et al. (2001) [11] introduced two applications regarding deniable authentication schemes: a coerced electronic voting system, and secure negotiations over the Internet. In this article, we focus on the first application and propose a protocol based on the Diffie-Hellman [12] algorithm.

A completed deniable authentication encryption scheme is defined as follows: *The integrity of a given message should be verified only by an intended receiver and the intended receiver cannot prove the source of the given message* *to any third party, even if he/she fully-cooperates with the third party.* As a result, we integrate the concept of deniable authentication into our protocol that enables a voting system to verify the source of a given ballot, but not prove to any third party the identity of the voter. The advantage of deniable authentication provides freedom from coercion in e-voting systems over insecure networks. Moreover, the proposed e-voting protocol with deniable authentication is based on the Diffie-Hellman algorithm. For the reason, the authentication of our e-voting protocol is deniable and no trusted third party is required [17].

# 3 The Proposed E-Voting Protocol

Motivated by the special characteristics of MANETs and based on the essential requirements of e-voting and the need to avoid the attacks mentioned in Section 1, in this section we demostrate a secure and efficient e-voting protocol with deniable authentication and apply it to ad hoc networks. Due to the absence of centralized authority, there is no need to be aided by a third party such as registration center, monitor center, or publishing center; and there are only two participants in the proposed e-voting protocol, namely voter and the system. Before describing our protocol, we define some basic notations and assumptions laid out in Table 1 and 2, respectively. The proposed e-voting protocol consists of two phases: the authentication phase and the voting phase. We describe the two phases in detail below.

## 3.1 Authentication Phase

The flowchart of this phase is shown in Figure 1. The chosen group leader (system) [29, 36], $S$ first generates an unique *tag#* for a vote and does the following:

**Step 1:** $S$ selects a random number, $a \in GF(p)$ to compute $X = g^a$ and

Table 1: Notations

| | |
|---|---|
| $(\mathbf{G}, g, p)$ | An element $g$ of large prime order $p$ in a finite cyclic group $\mathbf{G}$. |
| $V_i$ | The voter. |
| $S$ | The system. |
| $(pk_i, sk_i)$ | A public key and private key of node $i$, where $pk_i = g^{xi} = Y_i \in \mathbf{G}, sk_i = xi \in GF(p)$. |
| tag# | An unique tag number for a vote. |
| $\sigma_i^j$ | A signature on message $Msg_i^j$. |
| $VA_{pk_i}$ | An algorithm that on input $(m, \sigma)$, outputs 0 if $\sigma$ is not a valid signature of the message $m$ with respect to $pk_i$, and 1 otherwise. |
| $SK_i^j$ | A static shared key between node $N_i$ and node $N_j$, where $SK_i^j = Y_j^{xi} = g^{xixj} = Y_i^{xj} = SK_j^i$. |
| $m, m_i$ | The ballot and the marked ballot of $V_i$. |
| $H(.)$ | A collision-free one-way hash function. |
| $N_i$ | A nonce. |
| $E_{SK}[.]$ | The symmetric encryption function with shared session key $SK$. |
| $D_{SK}[.]$ | The symmetric decryption function with shared session key $SK$. |
| $E_{pk_i}(.)$ | The asymmetric encryption function with user $U_i$'s public key $pk_i$. |
| $D_{sk_i}(.)$ | the asymmetric decryption function with user $U_i$'s private key $sk_i$. |

Table 2: Assumptions

| | |
|---|---|
| A-1 | The links between connected mobile nodes are bidirectional and the mapping between the node $i$ identifier $ID_i$ and its corresponding public key is clear to every node that participates in the ad hoc network. |
| A-2 | Each node is capable of executing $E_{pk_i}(.)$, $D_{sk_i}(.)$, $E_{SK}[.]$, $D_{SK}[.]$ and $H(.)$ algorithms. |
| A-3 | The system is a trusted node and is responsible for supervising the whole procedure of voting and publishing the final voting result. |
| A-4 | The static shared key $SK_i^j$ is protected by node $U_i$ and node $U_j$ and the node never shares its private key with anyone else. |
| A-5 | An attacker may un-intrusively eavesdrop, modify or re-send messages into the wireless channel between two communication nodes in MANETs. However, we assumed that the proposed protocol does not provide a mechanism to against a Denial-of-Sevice (DOS) attack. For this attack, an attacker can simply disrupt, subvert, or destroy a network and this kind of attack is common though to all protocols in MANETs. |

makes ($Msg_S^1 = \{tag\#, ID_S, N_S, X, m\}, \sigma_S^1$) with his identity $ID_S$, a nonce $N_S$ and a blank ballot $m$, where $\sigma_S^1$ is $S$'signature on message $Msg_S^1$.

**Step 2:** $S$ broadcasts the message ($Msg_S^1 = \{tag\#, ID_S, N_S, X, m\}$, $\sigma_S^1$) to all voters.

**Step 3:** Each attended $V_i$ checks whether $VA_{pk_S}(Msg_S^1, \sigma_S^1) == 1$ holds or not. If it holds, $V_i$ selects a random number $b \in GF(p)$ to compute $Y = g^b$ and makes ($Msg_{V_i} = \{tag\#, ID_S, N_S, ID_{V_i}, N_{V_i}, Y\}, \sigma_{V_i}$) with his identity $ID_{V_i}$, a nonce $N_{V_i}$ and a signature $\sigma_{V_i}$ on message $Msg_{V_i}$; otherwise, stop.

**Step 4:** $V_i$ sends the message ($Msg_{V_i}, \sigma_{V_i}$) to $S$.

**Step 5:** $S$ checks the validity of $ID_{V_i}$, if the qualification and format are incorrect, the reply message will be rejected. Moreover, $S$ further checks whether $VA_{pk_vi}(Msg_{V_i}, \sigma_{V_i}) == 1$ holds or not. If it does not hold, stop; otherwise, $S$ computes $V_{S,V_i} = Y^a = g^{ab}$ and stores ($ID_{V_i}, N_{V_i}, V_{S,V_i}$) in his/her own database.

**Step 6:** Upon collecting all the received replies, $S$ makes ($Msg_S^2$=$\{tag\#, ID_S$, $N_S, H(ID_{V_i}, N_{V_i}, V_{S,V_i})\}$, $\sigma_S^2$) and broadcasts it to all voters.

**Step 7:** $V_i$ checks whether $VA_{pk_S}(Msg_S^2, \sigma_S^2) == 1$ holds or not. If it holds, $V_i$ computes $V'_{S,V_i} = X^b = g^{ab} = Y^a$ and checks whether $H(ID_{V_i}, N_{V_i}, V'_{S,V_i}) \stackrel{?}{=} H(ID_{V_i}, N_{V_i}, V_{S,V_i})$. If above holds, both the voter $V_i$ and the system $S$ can generate a unique and common vote ($ID_{V_i}, N_{V_i}, V_{S,V_i}$).

## 3.2 Voting Phase

**Step 1:** As the voter $V_i$ begins to vote, he/she makes their decision $m_i$, $challenge_i$ and computes $SK_{V_i}^S$, where $SK_{V_i}^S = Y_S^{xv_i} = g^{xsxv_i}$.

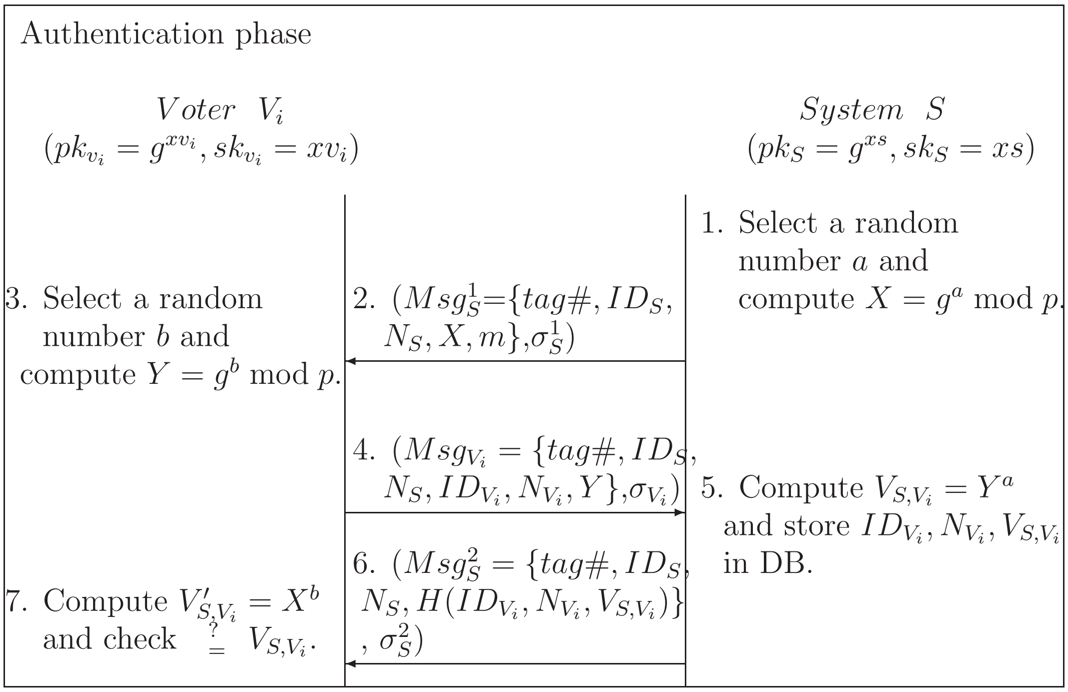


Figure 1: The data flow in the authentication phase

**Step 2:** $V_i$ sends $Msg_{V_i} = \{tag\#, ID_S, N_S, E_{pk_S}(ID_{V_i}, N_{V_i}, m_i, Challenge_i, H(ID_{V_i}, N_{V_i}, V_{S,V_i}, m_i, SK_{V_i}^S))\}$ to $S$.

**Step 3:** $S$ checks the validity of $tag\#$ and uses its private key $sk_V$ to recover $ID_{V_i}, N_{V_i}, m_i, Challenge_i$ and $H(ID_{V_i}, N_{V_i}, V_{S,V_i}, m_i, SK_{V_i}^S)$. Then, $S$ computes $SK_S^{V_i} = Y_{v_i}^{xs} = g^{xsxv_i} = Y_S^{xv_i}$ and checks whether $H(ID_{V_i}, N_{V_i}, V_{S,V_i}, m_i, SK_S^{V_i}) = H(ID_{V_i}, N_{V_i}, V_{S,V_i}, m_i, SK_{V_i}^S)$ holds or not. If it holds, $m_i$ is counted into the tally result and marks $(ID_{V_i}, N_{V_i}, V_{S,V_i})$ non-fresh; otherwise, stop.

**Step 4:** $S$ computes $E_{SK_S^{V_i}}[Challenge_i + 1]$ and sends it to $V_i$ for freshness checking.

**Step 5:** Finally, $V_i$ uses the static shared key $SK_{V_i}^S$ to recover message $Challenge_i + 1$ by computing $D_{SK_{V_i}^S}[E_{SK_S^{V_i}}[Challenge_i + 1]]$ for freshness checking. If it holds, $V_i$ is convinced that his/her vote has been counted into tally result.

| Voting phase | | |
|---|---|---|
| $Voter\ \ V_i$ $(pk_{v_i} = g^{xv_i}, sk_{v_i} = xv_i)$ | | $System\ \ S$ $(pk_S = g^{xs}, sk_S = xs)$ |
| 1. Select a $Challenge_i$ and compute $SK_{V_i}^S$. | 2. $Msg_{V_i}$=$\{tag\#, ID_S, N_S, E_{pk_S}(ID_{V_i}, N_{V_i}, m_i, Challenge_i, H(ID_{V_i}, N_{V_i}, V_{S,V_i}, m_i, SK_{V_i}^S))\}$. $\longrightarrow$ | |
| | | 3. Compute $SK_S^{V_i}$ and check $\overset{?}{=}$ $SK_{V_i}^S$. Mark $V_{S,V_i}$ non-fresh. |
| 5. Check $Challenge_i + 1$ | $\longleftarrow$ 4.$E_{SK_S^{V_i}}[Challenge_i + 1]$ | |

Figure 2: The data flow in the voting phase

# 4 Discussions and Analysis

In the section, we will analyze the essential requirements and security of our e-voting protocol and show a performance comparison of our protocol with other related protocols, in terms of the computational and communicative costs.

## 4.1 Requirements Analysis

In this subsection, we show that our proposed e-voting protocol meets the above-mentioned requirements of electronic voting listed in Section 1.1 and shown in Table 3.

**Completeness:** In our protocol, an attacker is unable to fake a vote unless he/she knows $V_{S,V_i}$ and $SK_{V_i}^S$. Furthermore, based on Assumption 3 mentioned in Section 3, without malicious system internally, no one can add an invalid ballot to the final tally, or remove a valid ballot from the final tally; and our protocol maintains the completeness of a voting system.

**Uniqueness:** For this requirement, it is important that a legal voter can not vote more than once. During the voting phase, the system will check whether the voter's status is fresh or not by comparing $ID_{V_i}, N_{V_i}, V_{S,V_i}$. If the system detects that the information $ID_{V_i}, N_{V_i}, V_{S,V_i}$ is marked as non-fresh, this indicates an occurrence of double voting. Consequently, a legal voter can cast at most one vote in our protocol.

**Privacy:** Public key techniques are what we employ in our protocol to satisfy the privacy requirement. During the voting phase, an attacker cannot obtain any knowledge from ballot messages in Step 2 because the voter encrypts the ballot messages with the system's public key $pk_S$. For this reason, no one can link the ballot to the voter due to absence of $Challenge_i$, $V_{S,V_i}$ and $SK^S_{V_i}$ and we have confirmed that our protocol meets this requirement.

**Eligibility:** Before casting a vote, a voter must register himself/herself to the system and the two sides must agree on vote information $V_{S,V_i}$ during the authentication phase. In addition, $V_i$ also needs the static shared key $SK^S_{V_i}$ to complete the vote during the voting phase. As a result, this requirement is achieved in our protocol.

**Fairness:** Due to the fact that an attacker cannot decrypt the ballot message without knowing the system's private key $sk_S$, no one can learn the partial results of an election and all the voted ballots are kept secret until the end of the voting session. Additionally, as a result of Assumption 3, the system cannot reveal the partial results to anyone before the end of voting and this requirement is confirmed in our protocol.

**Verifiability:** Based on the Assumption 3 mentioned in Section 3, our protocol uses a challenge-response concept during the voting phase and thus the voter $V_i$ can confirm that his/her vote has been counted correctly if $V_i$ verifies that the response from the system $S$ is valid. To do so, our protocol adds the voting response $E_{SK^{V_i}_S}[Challenge_i + 1]$ from $S$ to $V_i$ in order to achieve verifiability.

**Uncoercibility:** To prevent bribery and extortion, the voter $V_i$ cannot be able to prove to others for whom he/she has voted. Thus, $V_i$ is unable to reveal $m_i$ to others even if $V_i$ has been bribed and reveals $ID_{V_i}, N_{V_i}, V_{S,V_i}$ since $H(ID_{V_i}, N_{V_i}, V_{S,V_i}, m_i, SK^S_{V_i})$ is under the protection of the static shared key $SK^S_{V_i}$ during the voting phase. Therefore, our protocol achieves uncoercibility.

**Mobility:** Our protocol is designed to be implemented on mobile ad hoc networks and the voter can freely cast his/her marked ballot on the system through the ad hoc network without any restrictions on designated locations. Certainly, this requirement is met in our protocol.

**Deniable authentication:** According to the above mentioned attacks, we must show that the protocol is deniable. As the voter votes on a matter, he/she can deny he/she has voted to a third party since this secret information can be similarly generated by the system that also holds the same $V_{S,V_i}$ and $SK^S_{V_i}$. Moreover, in the voting phase, after receiving a vote $Msg_{V_i} = \{tag\#, ID_S, N_S, E_{pk_S}(ID_{V_i}, N_{V_i}, m_i, Challenge_i, H(ID_{V_i}, N_{V_i}, V_{S,V_i}, m_i, SK^S_{V_i}))\}$, the system can verify the source of $SK^S_{V_i}$ with his/her private key $sk_s$. But the system cannot prove the source of the vote $Msg_{V_i}$ to a third party. For $SK^S_{V_i} = Y_S^{xv_i} = g^{xsxv_i} = Y_{v_i}^{xs} = SK^{V_i}_S$, the system can make a marked ballot $m'_i$ and construct a vote $Msg'_{V_i} = \{tag\#, ID_S, N_S, E_{pk_S}(ID_{V_i}, N_{V_i}, m'_i, Challenge_i, H(ID_{V_i}, N_{V_i}, V_{S,V_i}, m'_i, SK^{V_i}_S))\}$, which is different from $Msg_{V_i}$. $Msg'_{V_i}$ is indistinguishable from the actual vote computed by $V_i$ and the system can simulate the vote of $V_i$. Hence, the third party cannot judge who has voted and this implies that our e-voting scheme is a deniable authentication protocol.

Table 3: Comparisons of general requirements between our protocol and other related protocols

| | Protocols | | | |
|---|---|---|---|---|
| Requirement | Dini's protocol [13] | Liaw's protocol [26] | Chang-Lee's protocol [6] | Our protocol |
| Completeness | Yes | Yes | Yes | Yes |
| Uniqueness | Yes | Yes | Yes | Yes |
| Privacy | Yes | Yes | Yes | Yes |
| Eligibility | Yes | Yes | Yes | Yes |
| Fairness | Yes | Yes | Yes | Yes |
| Verifiability | Yes | Yes | No | Yes |
| Uncoercibility | Yes | No | Yes | Yes |
| Mobility | Yes | Yes | Yes | Yes |
| Deniable authentication | No | No | No | Yes |

## 4.2 Security Analysis

In this subsection, we show how our protocol resists passive and active attacks as follows:

- Man-in-the-middle attack: If an attacker Eve wants to successfully inject fake secrets $X' = g^{a'}$ for voters and $Y' = g^{b'}$ for the system to perform the man-in-the-middle attack during the authentication phase, he must first be aware of the system's private key $sk_S$ and the voter's private key $sk_v$ used to generate signatures so as to convince both entities. In fact, she is unable to perform the man-in-the-middle attack in our protocol due to above-mentioned Assumption 4 in Section 3.

- Impersonation attack: To resist this attack, in our protocol, Eve is unable to impersonate a legal voter $V_i$ to cast a valid ballot unless she knows a secret $V_{S,V_i}$ and a shared key $SK_{V_i}^{S}$. However, based on Assumption 4, it is impossible for Eve to execute such an attack successfully in our protocol.

- Replay attack: In our protocol, this attack can be prevented by checking $tag\#$, nonces, and signatures during the authentication phase and Eve cannot forge the information $H(ID_{V_i}, N_{V_i}, V_{S,V_i}, m_i, SK^S_{V_i})$ to perform a replay attack in the voting phase.

- Eavesdropping attack: If Eve eavesdrops on the traffic between the voter and the system during the authentication phase, she can determine who communicates with the system. Although she ascertains $X = g^a$ and $Y = g^b$, from the computational Diffie-Hellman problem, she is unable to derive $V_{S,V_i} = g^{ab}$. Furthermore, she is unable to discover who the voter has voted for from the message transmitted in Step 2 and during the voting phase because it is protected by an one-way hash function. So, our protocol resists this kind of attack.

## 4.3 Performance Analysis

In this subsection, we evaluate the performance of the proposed e-voting protocol in terms of the total number of cryptographic operations performed during the authentication and voting phases, as shown in Table 4. For the communication round, multiple and independent messages can be sent in a single round. Owing to the requirement of scalability, too many interactions are not scalable in ad hoc networks. Therefore, the voter and the voting system must take the communication rounds into account during the whole election. In Table 5, we show the performance comparisons of our protocol with other that of related protocols reported [6, 8, 26, 35], in terms of computational and communicative costs.

From Table 5 shows that our proposed protocol, the computational loads of the voter and the system are almost equal because the system is also utilized via ad hoc nodes and the computational operations that are performed by the nodes should be balanced to meet the real circumstance of ad hoc

Table 4: Estimation of performance of the proposed e-voting protocol

| Phase | Voter $V_i$ | System $S$ | Communication cost |
|---|---|---|---|
| Authentication | $2T_{Exp}$+$1T_{Ha}$+ $1T_{En}$+$2T_{De}$ | $2T_{Exp}$+$1T_{Ha}$+ $2T_{En}$+$1T_{De}$ | 3 rounds |
| Voting | $1T_{Exp}$+$1T_{Ha}$+ $1T_{En}$+$1T_{Sym}$ | $1T_{Exp}$+$1T_{Ha}$+ $1T_{En}$+$1T_{Sym}$ | 2 rounds |

$T_{Exp}$: The number of exponentiation operation performed
$T_{Ha}$: The number of hashing operation performed
$T_{En}$: The number of asymmetric encryption operation performed
$T_{De}$: The number of asymmetric decryption operation performed
$T_{Sym}$: The number of symmetric encryption/decryption operation performed
Rounds: The number of communication rounds for voter and system

Table 5: Performance comparisons

| Protocol | Voter operations | System operations | Communication cost |
|---|---|---|---|
| Chang et al. [6] | $1T_{Exp}$+$2T_{Ha}$+ $1T_{En}$+$2T_{Sym}$ | $8T_{Exp}$+$3T_{De}$+ $23T_{Sym}$ | 4 rounds for voter 11 rounds for systems |
| Chen et al. [8] | $3T_{Exp}$+$1T_{Ha}$ | $7T_{Exp}$ | 3 rounds for voter 4 rounds for systems |
| Liaw [26] | $5T_{Exp}$+$1T_{Ha}$ | $5T_{Exp}$ | 2 rounds for voter 3 rounds for systems |
| Henríquez et al. [35] | $8T_{Exp}$+$3T_{En}$+ $2T_{De}$ | $2T_{Exp}$+$10T_{En}$+ $1T_{De}$ | 2 rounds for voter 2 rounds for systems |
| Our Protocol | $3T_{Exp}$+$2T_{Ha}$+ $2T_{En}$+$2T_{De}$+$1T_{Sym}$ | $3T_{Exp}$+$2T_{Ha}$+ $2T_{En}$+$2T_{De}$+$1T_{Sym}$ | 2 rounds for voter 3 rounds for system |

networks. Additionally, the most time-consuming operations are public key encryption $T_{En}$ and decryption $T_{De}$ and we can choose a lightweight public encryption/decryption algorithm to perform those operations. For example, ElGamal encryption [15] can be applied to our protocol. It is different from the RSA encryption [34] due to its security depending on the amount difficulty faced to solve discrete logarithm problem, as opposed to factoring problem. In ElGamal encryption, a sender $A$ computes $X_1 = g^r$ and $X_2 = m \oplus H(Y^r)$ and sends $(X_1, X_2)$ to a receiver $B$, where $r$ is a random number, $(x, Y = g^x)$ is $B$'s private/public key, H(.) is an one-way hash function and $m$ is a message. Afterwards, in ElGamal decryption, a receiver $B$ decrypts out the message $m$ by computing $m = X_2 \oplus H(C_1^x)$. From above computations, encryption and decryption can be roughly estimated as the following, $T_{En} = 2T_{Exp} + T_{Ha}$ and $T_{De} = T_{Exp} + T_{Ha}$, respectively. In addition, Elliptic Curve Cryptography (ECC) [23, 30] is widely being adopted to provide Public Key Cryptography (PKC) support in resource-constrained environments so that the existing PKC-based solutions can be exploited. Recently, TinyECC[18, 28], a software package, is being investigated to provide ECC-based PKC operations that can be flexibly configured and integrated into limited-resource sensor devices. Targeted at security of TinyECC, it provides PKC-based schemes that have proven to be secure and 160-bits ECC has the same security level as 1024-bits RSA. Moreover, TinyECC supports three well-known ECC mechanisms, including the Elliptic Curve Diffie-Hellman (ECDH) key agreement, the Elliptic Curve Digital Signature Algorithm (ECDSA), and the Elliptic Curve Integrated Encryption Scheme (ECIES). As a result, we can adopt TinyECC to provide a ready-to-use and publicly available software package for ECC-based public key cryptography operations in MANET applications and we believe that the performance of our proposed e-voting protocol is acceptable for mobile nodes and can be practically applied over MANETs.

## 5 Conclusion

In this article, we propose a deniable authentication e-voting protocol that not only meets the general requirements of a secure e-voting system but is also suitable for application over mobile ad hoc networks. The proposed protocol does not need the aid of centralized authorities such as authentication centers, tally centers, and monitor centers, which makes it more applicable than other existing protocols relying on on-line third party services. In the future, we plan to implement the proposed protocol into realistic scenarios to further evaluate the performance of our e-voting system through ad hoc networks.

## References


[1] Ahmed Abdel-Hafez, Ali Miri, and Louis Orozco-Barbosa, "Authenticated group key agreement protocols for ad hoc wireless networks," *International Journal of Network Security*, vol. 4, no. 1, pp. 90–98, 2007.

[2] Yonatan Aumann and Michael Rabin, "Authentication enhanced security and error correcting codes," in *Proceedings of the 18th Annual International Cryptology Conference on Advances in Cryptology*, pp. 299–303, Lecture Notes in Computer Science 1462, 1998.

[3] Yonatan Aumann and Michael Rabin, "Efficient deniable authentication of long messages," in *International Conference on Teoretical Computer Science in Honor of Professor Manuel Blum's 60th Birthday*, Hong Kong, China, 1998.

[4] F. Baiardi, A. Falleni, R. Granchi, F. Martinelli, M. Petrocchi, and A. Vaccarelli, "Seas, a secure e-voting protocol: Design and implementation," *Computers & Security*, vol. 24, no. 8, pp. 642–652, 2005.

[5] Azzedine Boukerche, Khalil El-Khatib, Li Xu, and Larry Korba, "An efficient secure distributed anonymous routing protocol for mobile and

wireless ad hoc networks," *Computer Communications*, vol. 28, no. 10, pp. 1193–1203, 2005.

[6] C. C. Chang and J. S. Lee, "An anonymous voting mechanism based on the key exchange protocol," *Computers & Security*, vol. 25, no. 4, pp. 307–314, 2006.

[7] D. Chaum, "Untraceable electronic mail, return addresses and digital pseudonyms," *Communications of the ACM*, vol. 24, no. 2, pp. 84–88, 1981.

[8] Yu-Yi Chen, Jinn-Ke Jan, and Chin-Ling Chen, "The design of a secure anonymous Internet voting system," *Computers & Security*, vol. 23, no. 4, pp. 330–337, 2004.

[9] Imrich Chlamtac, Marco Conti, and Jennifer J. N. Liu, "Mobile ad hoc networking: imperatives and challenges," *Ad Hoc Networks*, vol. 1, no. 1, pp. 13–64, 2003.

[10] L. Cranor and R. Cytron, "Sensus: A security-conscious electronic polling system for the Internet," in *Proceedings of the International Conference on System Sciences*, pp. 561–570, Hawaii, 1997.

[11] X. Deng, C. H. Lee, and H. Zhu, "Deniable authentication protocols," *IEE Proceedings Computers and Digital Techniques*, vol. 148, no. 2, pp. 644–654, 2001.

[12] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. IT-22, pp. 644–654, Nov. 1976.

[13] G. Dini, "A secure and available electronic voting service for a large-scale distributed system," *Future Generation Computer Systems*, vol. 19, no. 1, pp. 69–85, 2003.

[14] C. Dwork, M. Naor, and A. Sahai, "Concurrent zero-knowledge," in *Proceedings of the 30th ACM STOC'98*, pp. 409–418, Dallas, Texas, USA, 1998.

[15] T. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. IT-31, pp. 469–472, July 1985.

[16] L. Fan, C. X. Xu, and J. H. Li, "Deniable authentication protocol based on Diffie-Hellman algorithm," *Electronics Letters*, vol. 38, no. 14, pp. 705–706, 2002.

[17] Lei Fan, C. X. Xu, and J. H. Li, "Deniable authentication protocol based on deffie-hellman algorithm," *Eletronics Letters*, vol. 38, no. 14, pp. 705–706, 2002.

[18] TinyECC: A Configurable Library for Elliptic Curve Cryptography in Wireless Sensor Networks. http://discovery.csc.ncsu.edu/software/TinyECC/, 2007.

[19] Debasis Giri and Parmeshwary Dayal Srivastava, "An asymmetric cryptographic key assignment scheme for access control in tree structural hierarchies," *International Journal of Network Security*, vol. 4, no. 3, pp. 348–354, 2007.

[20] Fei Hu and Neeraj K. Sharma, "Security considerations in ad hoc sensor networks," *Ad Hoc Networks*, vol. 3, no. 1, pp. 69–89, 2005.

[21] Sheng-Yu Hwang, Hsiang-An Wen, and Tzonelih Hwang, "On the security enhancement for anonymous secure e-voting over computer network," *Computer Standards & Interfaces*, vol. 27, no. 2, pp. 163–168, 2005.

[22] J. Karro and J. Wang, "Towards a practical, secure, and very large scale online election," in *Proceedings of the 15th Annual Computer Security Applications Conference, ACSAC'99*, pp. 161–169, 1999.

[23] K. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.

[24] Nikos Komninos, Dimitris Vergados, and Christos Douligeris, "Detecting unauthorized and compromised nodes in mobile ad hoc networks," *Ad Hoc Networks*, vol. 5, no. 3, pp. 289–298, 2007.

[25] Cheng-Chi Lee, Li-Hua Li, and Min-Shiang Hwang, "A remote user authentication scheme using hash functions," *ACM Operating Systems Review*, vol. 36, no. 4, pp. 23–29, 2002.

[26] H. T. Liaw, "A secure electronic voting protocol for general elections," *Computers & Security*, vol. 23, no. 2, pp. 107–119, 2004.

[27] Iuon-Chung Lin, Min-Shiang Hwang, and Chin-Chen Chang, "Security enhancement for anonymous secure e-voting over a network," *Computer Standards & Interfaces*, vol. 25, no. 2, pp. 131–139, 2003.

[28] An Liu and Peng Ning, "Tinyecc: A configurable library for elliptic curve cryptography in wireless sensor networks," in *Proceedings of the 7th International Conference on Information Processing in Sensor Networks (IPSN 2008)*, 2008.

[29] N. Malpani, J. L. Welch, and N. Vaidya, "Leader election algorithms for mobile ad hoc networks," in *International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pp. 96–103, 2000.

[30] Victor S. Miller, "Use of elliptic curves in cryptography," in *Advances in Cryptology, CRYPTO'85*, pp. 417–426, Lecture Notes in Computer Science, Vol. 218, 1985.

[31] Ghassan Z. Qadah and Rani Taha, "Electronic voting systems: Requirements, design, and implementation," *Computer Standards & Interfaces*, vol. 29, no. 3, pp. 376–386, 2007.

[32] M. Ramkumar and N. Memon, "An efficient key predistribution scheme for ad hoc network security," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 3, pp. 611–621, 2005.

[33] Wei Ren, "Pulsing RoQ DDoS attacking and defense scheme in mobile ad hoc networks," *International Journal of Network Security*, vol. 4, no. 2, pp. 227–234, 2007.

[34] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[35] F. Rodríguez-Henríquez, Daniel Ortiz-Arroyo, and Claudia García-Zamora, "Yet another improvement over the Mu-Varadharajan e-voting protocol," *Computer Standards & Interfaces*, vol. 29, no. 4, pp. 471–480, 2007.

[36] G. Singh, "Leader election in complete networks," *SIAM Journal of Computing*, vol. 26, no. 3, pp. 772–785, 1997.

[37] Johann van der Merwe, Dawoud Dawoud, and Stephen Mcdonald, "A survey on peer-to-peer key management for mobile ad hoc networks," *ACM Computing Surveys*, vol. 39, no. 1, pp. 1–45, 2007.

[38] Yanchao Zhang, Wei Liu, Wenjing Lou, and Yuguang Fang, "Securing mobile ad hoc networks with certificateless public keys," *IEEE Transac-*

*tions on Dependable and Secure Computing*, vol. 3, no. 4, pp. 386–399, 2006.

[39] Lidong Zhou and Z. J. Haas, "Securing ad hoc networks," *IEEE Network*, vol. 13, no. 6, pp. 24–30, 1999.