A Taxonomy and Comparison of Remote Voting Schemes

Huian Li, Abhishek Reddy Kankanala, Xukai Zou
Department of Computer and Information Science
Purdue University Indianapolis, Indiana 46202, USA
Email: {huili, abhikank, xkzou}@cs.iupui.edu

***Abstract*—Remote voting has been an active research field for application of cryptographic techniques in the last two decades with many schemes and systems in publication. In this paper we present an overview of recent efforts in developing voting schemes and security models that involve a variety of real world constraints to ensure election integrity. We classify voting schemes based on their primary cryptographic techniques. We analyze recent typical schemes and systems against the basic and counter attack requirements with brief description. Such analysis shows difference among these security requirements and aids in design of future schemes. Our conclusion is provided regarding suitability of a particular voting system/scheme under various conditions.**

***Keywords*—Remote voting, Cryptography, Privacy, Electronic voting scheme, Protocol**

I. Introduction

Voting is the fundamental human right to voice opinion in democratic systems. From paper ballots and punch cards in early days, to current Direct-Recording Electronic (DRE) systems, the voting method has experienced tremendous transitions. With the advancement of cryptographic techniques and the new revolution in computing hardware, particularly, the portable electronic devices such as laptops, tablets, and smartphones, the voting techniques are embracing for an opportunity of dramatic improvement.

Many remote voting schemes and systems have been proposed in recent years to replace paper based voting for many obvious reasons such as efficiency and convenience. For example, remote voting allows people living in different places outside the country such as soldiers and diplomats to participate in elections easily by voting online. As pointed out in [16], the assumption of physical booths, hardware and software at polling places, and trusted human supervision in voting schemes, contradicts society's trend toward enabling interactions from anywhere at anytime.

In this paper, we investigate typical existing schemes and systems in literature, categorize them based on cryptographic techniques, and analyze unique techniques embedded in each of them. Our goal is to give a big picture of state-of-the-art voting methods and to enlighten us on the design of future voting schemes and systems.

Our paper is organized as follows. Section II presents common pieces in remote voting schemes, including entities, stages, and requirements. Section III lists typical remote voting assumptions and building blocks. Section IV describes categories of voting schemes based on primary cryptographic techniques. Section V gives the analysis of schemes with a comparison at the end. Section VI summarizes the paper.

II. Entities, Stages, and Requirements

In this section, we first provide an overview of typical entities and stages involved in a voting scheme. Then we give a list of voting requirements.

A. Typical Remote Voting Entities

Here we enumerate all entities that will be involved in the entire e-voting process. A typical e-voting scheme includes the following entities.

Voter. Voter is a person who will choose among the contesting candidates and cast vote. By using an authentication system, all schemes should be able to tell an eligible voter from ineligible ones. In schemes that considering voter coercion, a coercer could pretend to be a voter.

Candidate. Candidate is a person contesting in the elections. There are different positions to which a candidate may contest. The list of all the contestants or candidates will be shown to a voter and the voter has to decide for which candidate he has to vote.

Authority. Authority is an entity responsible for conducting the elections. The authority has to follow all the guidelines and implement the protocols for voting. Typically, a voting system will have multiple authorities.

Registrar. Registrar in a voting scheme is responsible for authenticating voters. Usually a set of registrars is assumed. They jointly issue keying materials such as private keys and public keys to voters.

Auditor. Auditor will audit the voting process by inspecting individual votes, final voting results, or voting logs.

Adversary. Apart from the entities above, there can be a malicious entity in the voting model called adversary which will attempt to manipulate the voting process and results. There are two types of adversary, *external* and *internal*. The external adversary will actively try to coerce a voter or breach the privacy of voters and an internal adversary, apart from breaching the privacy, may also try to corrupt the authority from inside. Adversary can also be classified into *passive* and *active*. The passive adversary honestly follow the protocol but try to infer more information, while the active adversary aims to violate the protocol in various ways.

Bulletin Board. This is the place publicly accessible to all entities listed above, usually with the appendive-write capability. It can be a public web site even allowing outsiders to access (but not write functionality). In certain schemes, it is a piece of universally accessible memory [31], [25].

However, not all schemes assume the existence of all entities mentioned above. For example, auditor is omitted from many schemes. In some schemes, authority may also carry the duty of registrar.

B. Generic Remote Voting Stages

In general, a voting scheme has several stages [59], [11] according to the voting model, and entities participate in different stages to follow cryptographic protocols and achieve requirements. The generic stages are given below.

Initialization. The authority or voting center prepares for the remote voting by following an initialization protocol. For example, in the RSA based scheme [14], the authority

978-1-4799-3572-7/14/$31.00 ©2014 IEEE

generates two large prime numbers as public and private keys for the rest of the voting process.

Registration/Authentication. Every voter has to register and be authenticated before casting vote. For example, the identification of a voter is checked using a browser based cryptographic protocol or by an email.

Vote Casting. The voter casts his vote using the exclusive credentials or keys given by the authority.

Vote Tallying. The authorities tally all the votes and publish the voting result. Typically votes are anonymous and sometimes encrypted.

Vote Verification. The voter should be provided with a verification procedure such that he can finally verify which candidate he has voted for.

Auditing. Auditing is done by the authority after finishing the voting procedure. The vote count is determined and will be sent to a central authority.

The above are generic stages followed by most of remote voting schemes. There might be schemes that will have some additional or less stages.

C. Major E-voting Requirements

In order for an e-voting scheme to be useful in practice, it should satisfy several requirements [55], [38], [21], [47], [56]. We enumerate the basic requirements first, and then give a list of requirements for countering attacks.

1) Basic Requirements: We argue that the requirements listed below are fundamental for every voting scheme.

Correctness. If all the participants are honest and following the protocol, the voting results reflect all voters' will.

Privacy. Privacy implies that voter's information should remain secret. This is actually one of voting regulations in certain countries. If a voter's information is published along with his vote, this certainly violates privacy.

Verifiability. A voter can verify his vote without revealing the identity and knows that his vote is counted (which is called individual verifiability). Anyone can verify the final voting results (which is called universal verifiability).

Eligibility. Only eligible voters are allowed to vote. This is usually implemented through authentication.

Reliability. The system should work without compromising votes, even if certain system failure occurs.

Transparency. Voters should be able to understand the system generally. If a voting scheme allows a voter to participate every stages from initial setup to final tally and verification, voters will be able to witness the whole process. Certainly this provides transparency, thus achieving voter assurance. In other words, voters will be assured that his vote is correctly cast and counted through transparency.

Fairness. Information about votes cannot be learned until the voting results are published. Any participants (even outsiders) cannot gain knowledge of the voting result before its final publication, thus precluding preannouncement of any partial voting results.

2) Counter Attack Requirements: It is important for a voting scheme to have certain counter attack requirements. Such features enhance the security of the model. A vulnerable voting scheme may be attacked by adversaries who will try to manipulate the election and lead to incorrect results. A secure voting scheme should possess the following requirements.

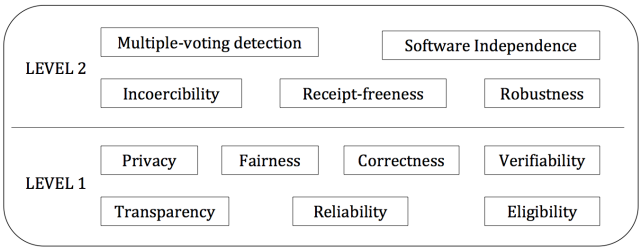

Fig. 1: Requirements of e-voting schemes

Robustness. Robustness of a scheme guarantees resistance against active and passive attacks, and any errors in the voting process. There might be several entities trying to disrupt the voting process, so a robust scheme must provide the required security and allow voters to complete voting without any interruption.

Incoercibility. In order to manipulate the voting results, an adversary may use multiple methods to coerce voters. For example, he can demand a voter to refrain from voting; or he can represent as a valid voter if he gets this voter's private key. So an incoercible voting scheme should defend against such adversary to coerce voters.

Receipt Freeness. A voter should not be provided with a receipt which may be used to trace the vote by other entity. In particular, a receipt may be used as a proof to show the vote is cast as being requested for vote buying and selling, and also for voter coercion.

Multiple-voting Detection. Multiple voting by a single voter will be identified. In a voting system allowing multiple votes, There are two approaches in literature [25], [31] to address this issue. One approach is that the last vote counts. In another approach, if all multiple votes by one voter devote to the same candidate, only one is counted. However, if these votes go to different candidates, all votes by this voter are cancelled. These counting approaches are usually used to defend voter coercion.

Software Independence. Software independence requires that an undetected change or error cannot cause an undetectable change or error in an election outcome [46]. This usually requires thorough evaluation and examination of the voting scheme.

Fig. 1 shows two levels of requirements of a voting scheme in which the first level are fundamental, while the second level includes counter attack and advanced requirements, and the requirements in higher level is more difficult to implement. In general, all the requirements are equally important for remote elections. However, depending on the voting situation and certain specific needs, some requirements may outweigh others. As pointed out in [25], there is not single voting scheme satisfying all requirements.

From a different perspective, several design principles for secure remote voting are listed in [53] including proven security, trustworthy design responsibility, published source code, vote verification, voter accessibility, ensuring anonymization, and expert oversight. Most of them correspond to the requirements mentioned above.

III. Assumptions and Building Blocks

In this section, we first give some common cryptographic assumptions. A few common building blocks are presented afterwards.

A. Common Assumptions

Untappable Channel. Several remote voting schemes, especially early ones, assume an untappable channel between communication parties, typically, voters and authorities (or registrars). It provides information-theoretic secrecy, but is not practical in reality. Certain scheme even makes stronger and unrealistic assumption such as an anonymous untappable channel.

Anonymous Channel. Anonymous channel is relaxed from untappable channel in terms of security, by allowing adversaries to spy on the communication channel and to intercept data. This is more realistic assumption about the distribution of credentials by registrars or authorities.

Voting Booth. Many remote voting schemes assume the existence of voting booth. The voting booths are governed by authorities, some even with guard. Usually only one voter is allowed to enter booth at a time and communication channels are provided between voting authorities and a booth so a voter can cast his vote. Typically no receipt is printed after a vote is casted. (In case a voter does receive a receipt, he will be asked to destroy it before leaving the booth.) The assumption provides receipt freeness. However, having physical booths is not a practical assumption for remote voting.

B. Common Building Blocks

Deniable Encryption. Deniable encryption is used against revealing encrypted information such that the owner of this information can decrypt it in an alternative way to different plaintext. It was introduced in [10], [9] that allows a sender to encrypt a bit b in such a way that the resulting ciphertext can be explained as either b or $1-b$ to a coercer.

Depending on which party being coerced, deniable encryption is classified into sender-deniable scheme (resilient against coercing, i.e., demanding to see the sender's ciphertext), receiver-deniable scheme, and sender-and-receiver-deniable scheme. Based on keys used between a sender and a receiver, it is also classified into public key deniable encryption and shared key deniable encryption. A sender-deniable public key encryption based on RSA was proposed in [47], and a receiver-deniable encryption scheme based on ElGamal was introduced in [34].

Zero-knowledge Proof. Zero-knowledge proof (ZKP) is also frequently used in various stages of a voting scheme between senders and receivers for verification purposes. It is a method by which one party (the prover) can prove to another party (the verifier) that a given statement is true, without conveying any additional information apart from the fact that the statement is indeed true. For cases where the ability to prove the statement requires some secret information on the part of the prover, the definition implies that the verifier will not be able to prove the statement to anyone else.

It has two forms, non-interactive ZKP and interactive ZKP. The first form is used primarily in voting schemes.

Secure Multi-party Computation. Secure multi-party computation is adopted into e-voting schemes to allow participants to carry computation jointly without anyone disclosing his data to others. Yao [58] has shown that any multi-party computation can be performed using a garbled circuit representation, but it is hard to implement efficiently.

Here we give secure two-party multiplication by Samet and Miri [49]. Initially, each of parties, M_i $(i=1,2)$, holds a *private* input x_i. At the end of the protocol, M_i will have a *private* output r_i, such that $x_1 \times x_2 = r_1 + r_2$. The protocol works as follows: 1) M_1 chooses a private key d and a public key e for an additively homomorphic public-key encryption scheme, with encryption and decryption functions being E and D, respectively. 2) M_1 sends $E(x_1,e)$ to M_2. 3) M_2 selects a random number r_2, computes $E(x_1,e)^{x_2}E(r_2,e)^{-1}$, and sends the result back to M_1. 4) M_1 decrypts the received value into $D(E(x_1,e)^{x_2}E(r_2,e)^{-1},d)$ and takes it as r_1.

Plaintext Equivalence Test. Plaintext equivalence test [30] is usually used to check if two encrypted votes are identical. Given two ciphertexts $\{v_1\}_k^{r_1}$ and $\{v_2\}_k^{r_2}$ respectively, with each encrypted using the same key k, a plaintext equivalence test allows the holders of the decryption key to demonstrate that plaintexts v_1 and v_2 are equal without revealing the decryption key or any information about v_1 and v_2.

There are other building blocks such as randomizer and commitment schemes. We omit the discussion due to space constraints.

IV. Primary Cryptographic Techniques and Corresponding Schemes

Here we present four primary cryptography techniques including mixnets, blind signature, threshold homomorphic encryption, and secret sharing, together with their corresponding e-voting schemes.

A. Mixnets Based Schemes

Mixnets is introduced by David Chaum [12] as a technique to provide anonymous communication, and later is applied into many voting schemes [3], [31], [36]. It is based on public key cryptography to provide anonymity and untraceability. Mixnets is multiparty communication protocol which takes input messages, shuffles them in a random order such that all the parties know that shuffling is performed and no party knows about shuffling algorithm.

In today's network, a sender and a receiver no longer remain confidential because every packet transmitted consists of IP addresses of the sender and the receiver. Any one can look through the packets to gain knowledge of the sender and the receiver. By using anonymous channels we can hide the information of sender so that even the receiver cannot relate back to the sender. In mixnets, such an anonymous communication is implemented by a set of nodes which take messages as inputs and bounce them back in a shuffled order [45].

When a sender wants to send a message, he passes it on to a node. Then the node permutes and passes the message to the next node, the last node sends the message to the receiver. As long as one node is honest and functions correctly, the anonymity of the sender can be guaranteed. Mixnets can be implemented in two categories:

1) Decryption Mixnets: In this type of mixnets, the nodes have a pair of public and private keys. A public key infrastructure is used to distribute the keys [3], [45]. Let pub_i be the public key and $priv_i$ the private key for the i-th node, and r_i be a random padding. The encryption protocol works as follows if a voter sends a message v through five nodes:

$$v_{enc} = E_{pub_1}(r_1, E_{pub_2}(r_2, E_{pub_3}(r_3, E_{pub_4}(r_4, E_{pub_5}(r_5, v)))))$$

Here the message will be encrypted in layers, the encrypted messages will pass through the nodes in the correct order, the nodes will decrypt the message and the last node delivers the message v. The decryption protocol works similarly by using the private keys.

2) Re-encryption Mixnets: Re-encryption mixnets also consists of multiple nodes to randomize and pass the messages. In this process, instead of decrypting the message from previous node, each node re-encrypts the message and passes on to the next node. Therefore if one node is honest we can guarantee that message is anonymous. The re-encryption mixnets can be deployed using different cryptosystems. One example is the ElGamal cryptosystem [3], [45].

B. Blind Signature Based Schemes

Blind signature [13] is applied in several e-voting schemes [39], [33], [44]. It is fundamentally one kind of digital signature where the content of a message is blinded before it is signed. In another word, it allows a person to get a third party to sign a message without revealing the content of a message, thus achieving confidentiality of the voter's ballot. Usually an authority blindly signs a voter's vote to authenticate it. Hence the authority whose function is to verify the eligibility of a voter will not know whom the voter votes for.

Currently blind key signature schemes are present with many public key protocols. One of such schemes is blind RSA scheme in which a traditional RSA signature is used. Here is a simple scheme of blind signature based on RSA signing [24]. Let (N,e) be the authority's public key and (N,d) be his private key where d is the inverse of $e \bmod \phi(N)$. A voter chooses a random number r such that $gcd(r,N) = 1$, and sends the following to the authority:

$$v' = v \cdot r^e \bmod N$$

The random number r is used to hide the ballot v from the authority. The authority then signs the blinded ballot after verification and sends back S'.

$$S' = (v')^d = v^d \cdot (r^e)^d = v^d \cdot r \bmod N$$

After receiving S', the voter unblinds it to get the true signature S since he knows r.

$$S = S' \cdot r^{-1} = v^d \cdot r \cdot r^{-1} = v^d \bmod N$$

Anonymous channels can be used to provide maximum privacy. A voter will get a blind signed vote, and then submit vote to the mixnets. After all the polls are completed, the mixnets will process the encrypted votes. The authority decrypts the votes shuffled by mixnets and displays the result to public. This approach is efficient but lacks transparency.

C. Threshold Homomorphic Encryption Based Schemes

Homomorphic encryption is used extensively among e-voting schemes [27], [6], [47]. It was first proposed by Benaloh et al. [5], [7]. It refers to a scenario in which the encryption of combined secret can be reconstructed from multiple independently encrypted secrets. Let the operations $\oplus$ and $\otimes$ be defined on the plaintext space and the ciphertext space, respectively. The "product" of the encryptions of two votes v_1 and v_2 is the encryption of the "sum" of two votes v_1 and v_2. More specifically, $E(v_1 \oplus v_2) = E(v_1) \otimes E(v_2)$. Examples of partially homomorphic cryptosystems [57] include ElGamal, Paillier, RSA, and a few others. Below we give further detail of ElGamal and Paillier cryptosystems.

1) ElGamal: The ElGamal cryptosystem [20] is adopted often in e-voting schemes. It is by nature homomorphic with multiplication. Assume in a commutative group $\mathbb{G}$ of order $|G| = q$, the public key is $(\mathbb{G},q,g,h)$ where g is a generator of G, $h = g^x$, and x being the secret key. The encryption of a vote v is $E(v) = (\alpha,\beta) = (g^r, v \cdot h^r)$ for some random $r \in \{0,1,...,q-1\}$. For two votes which are encrypted as

$$\begin{aligned} E(v_1) &= (\alpha_1,\beta_1) = (g^{r_1}, v_1 \cdot h^{r_1}), \text{ and} \\ E(v_2) &= (\alpha_2,\beta_2) = (g^{r_2}, v_2 \cdot h^{r_2}) \end{aligned}$$

The homomorphic property is then:

$$\begin{aligned} E(v_1) \cdot E(v_2) &= (\alpha_1,\beta_1) \cdot (\alpha_2,\beta_2) \\ &= (g^{r_1}, v_1 \cdot h^{r_1}) \cdot (g^{r_2}, v_2 \cdot h^{r_2}) \\ &= (g^{r_1+r_2}, (v_1 \cdot v_2) h^{r_1+r_2}) = E(v_1 \cdot v_2) \end{aligned}$$

The encrypted votes are "summed" by using the homomorphic property of the encryption function (without decrypting them). However, ElGamal is only multiplicative homomorphic as shown above, so e-voting systems taking advantage of ElGamal usually adopt a slightly modified cryptosystem which is additive homomorphic.

The modified ElGamal works as below. The encryption of a vote v is $E(v) = (\alpha,\beta) = (g^r, p^v \cdot h^r)$ where p is another independent (from g) generator of $\mathbb{G}$. For two votes which are encrypted as

$$\begin{aligned} E(v_1) &= (\alpha_1,\beta_1) = (g^{r_1}, p^{v_1} \cdot h^{r_1}), \text{ and} \\ E(v_2) &= (\alpha_2,\beta_2) = (g^{r_2}, p^{v_2} \cdot h^{r_2}) \end{aligned}$$

So the additive homomorphic property is:

$$\begin{aligned} E(v_1) \cdot E(v_2) &= (\alpha_1,\beta_1) \cdot (\alpha_2,\beta_2) \\ &= (g^{r_1}, p^{v_1} \cdot h^{r_1}) \cdot (g^{r_2}, p^{v_2} \cdot h^{r_2}) \\ &= (g^{r_1+r_2}, p^{v_1+v_2} h^{r_1+r_2}) = E(v_1+v_2) \end{aligned}$$

2) Paillier: The Paillier cryptosystem [40] is another probabilistic encryption function for public key cryptography. The notable feature is its homomorphic property. In this cryptosystem, assume the public key is the modulus m and the base is g, the encryption of a vote v is $E(x) = g^x r^m \bmod m^2$ for some random $r \in \{0,1,...,m-1\}$. For two votes which are encrypted as

$$\begin{aligned} E(v_1) &= g^{v_1} r_1^m, \text{ and} \\ E(v_2) &= g^{v_2} r_2^m \end{aligned}$$

The homomorphic property is then:

$$\begin{aligned} E(v_1) \cdot E(v_2) &= (g^{v_1} r_1^m)(g^{v_2} r_2^m) \\ &= g^{v_1+v_2} (r_1 r_2)^m = E(v1+v2 \bmod m) \end{aligned}$$

In many voting schemes, all votes can be added by using homomorphic encryption without decrypting them first, therefore ensuring privacy. So homomorphic encryption is an efficient method for e-voting schemes but it cannot be applied to schemes in which addition is not performed as a main method for combination.

D. Secret Sharing Based Schemes

Secret sharing is also used in many e-voting schemes due to its efficiency and simplicity. In Shamir's scheme [19], a secret s in a finite field is partitioned into n shares where any k-sized subset of these n shares reveals s ($k \leq n$), but any subset of size smaller than k reveals nothing about s.

A secret sharing scheme can have the property of homomorphism. As an example, simplified (n,n) secret sharing [60]

TABLE I: Typical cryptographic primitives in e-voting and their comparison

Typical Crypto-primitives	Advantages	Disadvantages
Mixnets	Shuffling makes votes unlinkable to voters; No fixed sequence of stages is required [50], [22]	Multiple encryptions are needed for input; Large size messages are not efficiently accommodated [50], [22]
Blind signature	Efficient and simple to implement	Signer has no control over attributes except for those bound by public key; Universal verifiability is hard to implement
Homomorphic encryption	Tallying procedure is simple; Votes cannot be tallied before being cast	Susceptible to some attacks such as RSA blinding attack; Concern over zero knowledge proof used in voting schemes
Secret sharing	Increased reliability and confidentiality; Low security demand of communication channel	Robust but harder to implement; Concern over scalability

is additively homomorphic. Here, the secret s is partitioned into n shares s_i $(1 \leq i \leq n)$ such that $s = \sum_{i=1}^{n} s_i$. The sum of two shares s_i and s_i' (corresponding to s and s' respectively) is a share of the sum of the two secrets s and s'.

Several voting schemes [37], [18], [28] exploit homomorphism based on secret sharing. Some schemes [18] utilize Shamir's threshold secret sharing, while some [28] are based on Chinese remainder theorem.

Table I summarizes the four major cryptographic techniques used in e-voting systems. Currently, most schemes utilize mixnets and homomorphic encryption.

V. Analysis of Existing Schemes

In this section, we give analysis of several remote voting schemes. For each scheme, a short description is provided, together with main properties.

A. UVote

Category: Mixnets

Description: In this scheme [1], voters are required for pre-registration and have the ability to vote multiple times. Only the last vote is counted and higher priority will be given to the vote casted from a voting center because the voter is personally present to verify his details, which makes the system coercion resistant. Vote selling is not possible because a voter can later change his vote by casting from another device.

Voters initially register their mobile numbers or email addresses as primary account, and can add multiple accounts later. The primary account is used for coercion resistance. Any notifications and messages are sent to the primary account for verification and it cannot be deleted online. Voters get their public and private key pairs for encryption and decryption of votes. A unique PIN is provided to access electoral website.

Properties: Verifiability is achieved through providing confirmation messages and notices. Universal verifiability is ensured by the back end system. No partial result or individual vote is revealed until the end of election, thus achieving fairness. Receipt freeness is not achieved as a receipt is provided to the voter.

B. Zeus

Caterogy: Mixnets

Description: Voters in the web based Zeus system [54] need to visit the web site to register public and private keys. The browser verifies a key by comparing the hash value with the registered one. This scheme uses the same cryptographic techniques as Helios [4]. Mixing is done by the Zeus system and external agents. Trustees are notified for decryption after the mixing process is finished. The Zeus system combines the decryptions and produces results. The results may be presented using external algorithms.

Properties: The results are posted onto bulletin board for universal verification. Receipt freeness in not achieved as a cryptographically signed receipt is provided to the voter at the end.

C. Cobra

Category: Homomorphic encryption

Description: In this Scheme [23], voters have to register by creating and submitting an encrypted credential. This credential is homomorphically added to an encrypted Bloom Filter[43], [29]. A voter selects certain number of candidate passwords and registers one of them. Then the voter encrypts the vote and uses his password to regenerate credential. A coercer is not able to manipulate a voter's vote because he may give the coercer a fake password or panic password [15]. Ballots are submitted using anonymous channels. Tallying is performed by the authority. The ballots are homomorphically added and decrypted, and results are published.

Properties: The final tally is verifiable. This scheme is coercion resistant as it uses a fake credential feature.

D. Helios

Category: Mixnets, homomorphic encryption

Description: Helios is a web based voting scheme [4]. It runs as a client program in browser, so voters can use their browsers to submit votes. Helios' protocol is similar to Benaloh's simple verifiable voting protocol [35]. mixnets is the tool to anonymize ballots.

Properties: A voter can verify that his ballot is received correctly. It also provides universal verifiability. Anyone can independently and externally verify that all votes were valid without any credentials. It is meant for low coercion elections.

E. Secure Internet Voting Using DSA Public Keys

Category: Mixnets

Description: This scheme [26] depends on anonymity provided by shuffling Digital Signature Algorithm (DSA) public keys. The voting credentials consist of simple DSA public keys. The shuffling of these keys leads to a list of anonymous keys, which can no longer be attributed to individual voters, but can still be used to verify their signatures.

Properties: This system offers universal verifiability. Receipt freeness is also achieved. However, coercion resistance is questionable.

F. Civitas

Category: Mixnets

Description: Civitas [16] utilizes a publicly viewable log service such as a bulletin board, and integrity is maintained by using digital signatures. Protocol compliance is enforced through several instances of ZKP. In order to resist coercion, a voter has to use his designated private key and run an algorithm to generate fake credentials. The voter provides these fake credentials in case of coercion from adversaries. All the votes submitted through fake credentials are eliminated, and the re-voting depends on the policy specified by the supervisor. An enhanced scheme [52] introduces robustness into Civitas.

Properties: Tabulation is publicly verifiable. Additional verifiability is ensured through log records. Coercion resistance is achieved by giving away fake credentials.

G. Multi-Authority E-voting System

Category: Homomorphic encryption

Description: This scheme [2] consists of multiple authorities, i.e., the election is controlled by multiple administrators to ensure privacy and overcome colluding in case of single administrator. This scheme uses homomorphic encryption. A voter's ballot is digitally signed with ElGamal DSA and encrypted with the additive ElGamal scheme.

Properties: Several security properties, such as fairness and completeness, are guaranteed. Privacy is achieved using encryption. ZKP is used through out the process. Voters are allowed to vote only once. However, this property can be used by adversaries to vote on their behalf. A voter has no option of giving away a fake vote in order to defend against coercion.

H. E-NOTE

Description: Two levels of security measure are used in this E-NOTE scheme [42] to prevent leakage of privacy or collusion of authorities. E-NOTE is an enhanced version of NOTE (Name and vOte separaTed E-voting scheme) [41] which separates names and votes on ballots so that privacy concerns during vote counting can be eliminated. All voting transactions are recorded in order to prevent fraud.

A voter has to register through an election authority first and get a certificate. The voter uses this certificate to obtain ballot from a ballot distribution center. This method ensures confidentiality as there is no linkage between voter's true identity and his certificate. the electronic ballot consists of three parts. It is sent to the vote counting committee (VCC), but VCC can decrypt only one part which contains the voting data. Tallying is done without matching the voter's name with his vote. This ensures the privacy and can also protect him from adversaries as they do not know which person has voted for which candidate.

Properties: Each voter obtains a watchdog device from the election commission to achieve confidentiality. Coercion resistance property is not given importance in this scheme. A voter is given a receipt to review and track his vote, so receipt freeness is not achieved.

I. Bingo Voting

Description: This scheme [8] provides an efficient way to achieve verifiability and coercion resistance based on a trusted random number generator. A bulletin board is used for recording and public verifiability. Every ballot has a unique serial number and each voter can take a copy with them after voting. Every voter will be given a receipt for all the candidates, even those he did not vote for. All serial numbers are printed on the same receipt.

Dummy votes are generated for every candidate and are shuffled before election. These fake or dummy votes are also published in the bulletin board so a voter can use them to avoid vote selling pressure from adversary. The tallying takes place by only counting the real votes. Dummy votes can be distinguished as those random numbers are taken from a pool which was created before. Here the voter is assured that the original vote is tallied. A cryptographic proof is submitted and can be checked after tallying. In the final stage only, the original votes are counted and results are displayed.

Properties: Voter has ability to check his vote, achieving verifiability. This scheme ensures privacy and correctness. Coercion resistance is effectively implemented. Receipt freeness is also achieved since the given receipt cannot prove to others which candidate the voter has voted for.

J. VoteBox

Description: The VoteBox system [51] utilizes a distributed broadcast network and replicated log, providing robustness and auditability in case of failure, misconfiguration, or tampering. The system utilizes an immediate ballot challenge to assure a voter that his ballot is cast as intended. Additionally, the vote decryption key can be distributed to several mutually-untrusted parties. VoteBox provides strong auditing functionality.

Properties: The scheme is receipt free and allows voter to verify his vote. It assumes the existence of voting booth, so coercion resistance and privacy are achieved since an adversary cannot enter the booth with the voter.

K. Prêt à Voter

Category: Mixnets

Description: The Prêt à Voter scheme [48] encodes a voter's vote using a randomized candidate list. The randomization ensures the secrecy of the voter's vote. After a voter casts vote in a voting booth, he is given a receipt such that the voter can verify his receipt appears on the bulletin board. The receipt in an encrypted form can be used to reconstruct the candidate order and consequently the vote he has cast. In reality, the secret keys of the encryption are shared across several tellers. All votes are published on a bulletin board for verification by voters. The tellers will take all submitted receipts, apply mix networks, and then decrypt them to tally votes.

Properties: This scheme provides end-to-end verifiability and receipt freeness. With the assumption of voting booth, it achieves privacy and coercion resistance.

L. ADDER

Category: Homomorphic encryption

Description: Adder [32] is an open source Internet based voting system. A public key is set up for the voting system, and a private key is shared by a set of authorities. Each voter encrypts his vote using the public key. The encrypted vote and its ZKP are published onto the bulletin board. Due to the homomorphic property, the encrypted tally is obtained by multiplying all encrypted votes on the bulletin board. The authorities then work together to obtain the decrypted tally. The trust is split among authorities.

Properties: This scheme provides universal verifiability. Depending on the voting setup (such as Internet or polling places), it may allow verification by an individual voter. Each procedure is supervised by multiple authorities, and the final tally cannot be revealed without the cooperation of a given number of authorities, so the trust is distributed.

M. Scytl Remote Voting System

Category: Secret sharing, Mixnet

Description: Scytl [17] is an Internet-based voting system currently in operation. Basically, it exploits secret sharing to distribute a voter's private key to many servers (and destroy the key after distribution). It assumes components such as voting kiosk located in polling place and SSL connection over VPN between polling centers and election centers. Mixing protocols are followed after voting. Scytl copies all collected data to appropriate backup media and delivers the data to authorities for future auditing.

Properties: This scheme has several assumptions. Vote verification is provided. Scytl's coercion resistance is based on its architecture.

TABLE II: Comparison of voting schemes based on the requirements

Schemes/ Systems	Individual Verifiability	Universal Verifiability	Fair-ness	Coercion-resistance	Robust-ness	Receipt freeness	Cryptographic primitives	Tailored Hardware
UVote [1]	Y	NK	NK	Y	NK	N	Mixnets	N
Zeus [54]	Y	Y	Y	N	Y	N	Mixnets, ZKP	N
Cobra [23]	N	N	Y	Y	Y	Y	HE, EBF	N
Helios [4]	Y	Y	Y	N	Y	N	Mixnets, ZKP	N
DSA Public Keys [26]	Y	Y	Y	Y	Y	NK	HE, Mixnets, ZPK	N
RSA Puzzle Lock [14]	Y	NK	Y	NK	Y	NK	RSA	N
Civitas [16]	Y	NK	Y	Y	Y	Y	Mixnets	N
Multi-Authority [2]	NK	Y	Y	Y	Y	Y	HE, ElGamal DSA	N
E-NOTE [42]	Y	Y	Y	NK	Y	N	RSA	Watchdog
Bingo [8]	Y	Y	Y	Y	Y	Y	CM, ZKP	Booth
VoteBox [51]	Y	Y	NK	Y	Y	Y	HE, HC	Booth
Prêt à Voter [48]	Y	Y	Y	Y	Y	Y	Mixnets	Booth
ADDER System [32]	NK	Y	Y	Y	Y	Y	HE, ZKP	N
Scytl [17]	Y	Y	Y	Y	Y	Y	SS, Mixnets	Booth, VM

Y: Yes; N: No; NK: Not Known; HE: Homomorphic Encryption; EBF: Encrypted Bloom Filter; ZKP: Zero Knowledge Proof; CM: Commitment; HC: Hash Chaining; SS: Secret Sharing; VM: Verification Module; Booth: Voting Booth

Table II gives a comprehensive view of the requirements satisfied by the schemes mentioned above, together with their corresponding cryptographic techniques and necessary hardware. We have to point out that

- All schemes claim to achieve privacy, eligibility, and correctness, so we leave out these columns in the table.
- We omit software independence from the table simply because it is not clearly indicated in each scheme, and hard to evaluate.
- For a given requirement, some schemes may claim success in their implementations, but the degree of success varies. Take robustness as an example, a few schemes assume that one voter can vote once only. With this assumption, it certainly excludes some vulnerabilities, while others can handle this particular case.
- The definition of requirements in each scheme/system is also not identical. For example, in Uvote [1], an SMS message is considered as individual verification.
- Several schemes claimed coercion resistance. However, their implementations are quite different. Some of them take advantage of fake/panic passwords, while others assume the voting booth because the coercer cannot enter it together with the voter.

VI. Conclusion

In this paper, we present a comprehensive review of the recent research work on the remote voting schemes and systems. We discuss the voting requirements and cryptographic building blocks, and categorize the schemes based on cryptographic techniques. We also study in-depth the security properties of each scheme. Our observations are as follows.

1) There is not a scheme or system satisfying all requirements.
2) Currently, coercion resistance relies on the fake/panic credential or polling places. However, the assumption of a polling place is not suitable and realistic for remote voting.
3) Transparency is not implemented in most of schemes. Even though some schemes claims to be transparent, it is still vague to voters. We argue transparency is important in a voting scheme for giving assurance to voters. In return, it can win voter's confidence in the voting process.
4) The involvement of special device can potentially be a burden to voters, since one primary goal of remote voting is to provide convenience to voters.

With our analysis, we hope to shed some light on the directions of future remote voting design.

References

[1] R. Abdelkader and M. Youssef. Uvote: A ubiquitous e-voting system. In *Mobile, Ubiquitous, and Intelligent Computing (MUSIC), 2012 Third FTRA International Conference on*, pages 72–77, 2012.

[2] A. P. Adewole, A. S. Sodiya, and O. A. Arowolo. A receipt-free multi-authority e-voting system. *Int. Jour. of Comp. App.*, 30(6):15–23, Sep. 2011.

[3] B. Adida. *Advances in Cryptographic Voting Systems*. PhD thesis, Cambridge, MA, USA, 2006. AAI0810143.

[4] B. Adida. Helios: Web-based open-audit voting. In *Proceedings of the 17th Conference on Security Symposium*, SS'08, pages 335–348, 2008.

[5] J. Benaloh. Secret sharing homomorphisms: Keeping shares of a secret secret (extended abstract). In *Advances in Cryptology - CRYPTO'86*, Lecture Notes in Computer Science, pages 251–260. Springer Berlin / Heidelberg, 1987.

[6] J. Benaloh. *Verifiable Secret Ballot Elections*. PhD thesis, Yale University, 1987.

[7] J. Benaloh and M. Yung. Distributing the power of a government to enhance the privacy of voters. In *Proceedings of the fifth annual ACM symposium on Principles of distributed computing*, PODC '86, pages 52–62, New York, NY, USA, 1986. ACM.

[8] J. Bohli, J. Müller-Quade, and S. Röhrich. Bingo voting: Secure and coercion-free voting using a trusted random number generator. In *Proceedings of the 1st International Conference on E-voting and Identity*, VOTE-ID'07, pages 111–124, 2007.

[9] R. Canetti, C. Dwork, M. Naor, and R. Ostrovsky. Deniable encryption. In Burton Kaliski, editor, *Advances in Cryptology CRYPTO '97*, volume 1294 of *Lecture Notes in Computer Science*, pages 90–104. Springer Berlin / Heidelberg, 1997.

[10] R. Canetti and R. Gennaro. Incoercible multiparty computation. In *Proceedings of the 37th Annual Symposium on Foundations of Computer Science*, FOCS '96, pages 504–, Washington, DC, USA, 1996. IEEE Computer Society.

[11] O. Cetinkaya and A. Doganaksoy. A practical verifiable e-voting protocol for large scale elections over a network. In *Availability, Reliability and Security, 2007. ARES 2007. The Second International Conference on*, pages 432–442, 2007.

[12] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24(2):84–90, February 1981.

[13] D. Chaum. Blind signatures for untraceable payments. In *Advances in Cryptology: Proceedings of CRYPTO '82*, pages 199–203. Plenum, 1982.

[14] H. Chen and R. Deviani. A secure e-voting system based on rsa time-lock puzzle mechanism. In *The 7th International Conference on BWCCA*, pages 596–601, 2012.

[15] J. Clark and U. Hengartner. Panic passwords: Authenticating under duress. In *Proceedings of the 3rd Conference on Hot Topics in Security*, HOTSEC'08, pages 8:1–8:6, 2008.

[16] M. Clarkson, S. Chong, and A. Myers. Civitas: Toward a secure voting system. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, pages 354–368, 2008.

[17] M. Clarkson, B. Hay, M. Inge, D. Wagner, and A. Yasinsac. Software review and security analysis of scytl remote voting software, September 2008.

[18] R. Cramer, R. Gennaro, and B. Schoenmakers. A secure and optimally efficient multi-authority election scheme. In *Proc. of the 16th annual Int. Conf. on Theory and App. of cryptographic techniques*, EUROCRYPT'97, pages 103–118, 1997.

[19] E. Dawson and D. Donovan. The breadth of shamir's secret-sharing scheme. *Comput. Secur.*, 13(1):69–78, February 1994.

[20] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In George Blakley and David Chaum, editors, *Advances in Cryptology*, volume 196 of *Lecture Notes in Computer Science*, pages 10–18. Springer Berlin / Heidelberg, 1985.

[21] J. Epstein. Electronic voting. *IEEE Computer*, 40(8):92–95, 2007.

[22] J. Esch. Prolog to a survey on mix networks and their secure applications. *Proceedings of the IEEE*, 94(12):2139–2141, 2006.

[23] A. Essex, J. Clark, and U. Hengartner. Cobra: toward concurrent ballot authorization for internet voting. In *Proceedings of the 2012 international conference on Electronic Voting Technology/Workshop on Trustworthy Elections*, EVT/WOTE'12, pages 3–3, Berkeley, CA, USA, 2012. USENIX Association.

[24] S. Goldwasser and M. Bellare. Lecture notes on cryptography. *Summer course Cryptography and computer security at MIT*, 1999.

[25] G. S. Grewal, M. D. Ryan, S. Bursuc, and P. Y. Ryan. Caveat coercitor: coercion-evidence in electronic voting. In *IEEE Security and Privacy Symposium*, 2013.

[26] R. Haenni and O Spycher. Secure internet voting on limited devices with anonymized dsa public keys. In *Proceedings of the 2011 Conference on Electronic Voting Technology/Workshop on Trustworthy Elections*, EVT/WOTE'11, 2011.

[27] M. Hirt and K. Sako. Efficient receipt-free voting based on homomorphic encryption. In *Proc. of the 19th Int. Conf. on Theory and App. of cryptographic techniques*, pages 539–556, 2000.

[28] S. Iftene. General secret sharing based on the chinese remainder theorem with applications in e-voting. *Electron. Notes Theor. Comput. Sci.*, 186, July 2007.

[29] W. Itani, C. Ghali, A. El Hajj, A. Kayssi, and A. Chehab. Sinpack: A security protocol for preventing pollution attacks in network-coded content distribution networks. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–6, 2010.

[30] M. Jakobsson and A. Juels. Mix and match: Secure function evaluation via ciphertexts. In Tatsuaki Okamoto, editor, *Advances in Cryptology ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Computer Science*, pages 162–177. Springer Berlin / Heidelberg, 2000.

[31] A. Juels, D. Catalano, and M. Jakobsson. Coercion-resistant electronic elections. In *Proceedings of ACM workshop on Privacy in the electronic society*, WPES '05, pages 61–70, New York, NY, USA, 2005.

[32] A. Kiayias, M. Korman, and D. Walluck. An internet voting system supporting user privacy. In *Proceedings of the 22Nd Annual Computer Security Applications Conference*, ACSAC'06, pages 165–174, Washington, DC, USA, 2006. IEEE Computer Society.

[33] K. Kim, J. Kim, B. Lee, and G. Ahn. Experimental design of worldwide internet voting system using pki. In *Proc. of SSGRR Int. Conf. on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet*, SSGRR2001, 2001.

[34] M. Klonowski, P. Kubiak, and M. Kutyłowski. Practical deniable encryption. In Viliam Geffert, Juhani Karhumäki, Alberto Bertoni, Bart Preneel, Pavol Návrat, and Mária Bieliková, editors, *SOFSEM 2008: Theory and Practice of Computer Science*, volume 4910 of *Lecture Notes in Computer Science*, pages 599–609. Springer Berlin / Heidelberg, 2008.

[35] R. Kusters and T. Truderung. An epistemic approach to coercion-resistance for electronic voting protocols. In *Security and Privacy, 2009 30th IEEE Symposium on*, pages 251–266, 2009.

[36] B. Lee, C. Boyd, E. Dawson, K. Kim, J. Yang, and S. Yoo. Providing receipt-freeness in mixnet-based voting protocols. In *In Proc. of Information Security and Cryptology (ICISC03), volume 2971 of LNCS*, pages 245–258. Springer, 2003.

[37] B. Lee and K. Kim. Receipt-free electronic voting through collaboration of voter and honest verifier. In *Proc. of JW-ISC2000*, pages 101–108, 2000.

[38] C. Li and M. Hwang. Security enhancement of chang-lee anonymous e-voting scheme. *International Journal of Smart Home*, 6(2):45–52, April 2012.

[39] M. Ohkubo, F. Miura, M. Abe, A. Fujioka, and T. Okamoto. An improvement on a practical secret voting scheme. In *Proceedings of the Second International Workshop on Information Security*, ISW '99, pages 225–234, London, UK, UK, 1999. Springer-Verlag.

[40] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In Jacques Stern, editor, *Advances in Cryptology EUROCRYPT 99*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238. Springer Berlin / Heidelberg, 1999. 10.1007/3-540-48910-X_16.

[41] H. Pan, E., Hou, and N. Ansari. Ensuring voters and candidates' confidentiality in e-voting systems. In *34th IEEE Sarnoff Symposium*, May 2011.

[42] H. Pan, E. Hou, and N. Ansari. E-note: An e-voting system that ensures voter confidentiality and voting accuracy. In *IEEE International Conference on Communications (ICC)*, pages 825–829, June 2012.

[43] H. Perl, Y. Mohammed, M. Brenner, and M. Smith. Fast confidential search for bio-medical data using bloom filters and homomorphic cryptography. In *E-Science (e-Science), 2012 IEEE 8th International Conference on*, pages 1–8, 2012.

[44] M. Radwin. An untraceable, universally verifiable voting scheme. *Seminar in Cryptology*, 1995.

[45] P. Ribarski and L. Antovski. Mixnets: Implementation and performance evaluation of decryption and re-encryption types. In *Information Technology Interfaces (ITI), Proceedings of the ITI 2012 34th International Conference on*, pages 493–498, 2012.

[46] R. L. Rivest. On the notion of software independence in voting systems. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881):3759–3767, 2008.

[47] Z. Rjaskova. *Electronic Voting Schemes*. PhD thesis, Comenius University, 2002.

[48] P. Y. A. Ryan, D. Bismark, J. Heather, S. Schneider, and Z. Xia. Prêt à voter: a voter-verifiable voting system. *Information Forensics and Security, IEEE Transactions on*, 4(4):662–673, Dec 2009.

[49] S. Samet and A. Miri. Privacy preserving ID3 using Gini index over horizontally partitioned data. In *Proc. of the 2008 IEEE/ACS*, AICCSA '08, pages 645–651, Washington, DC, USA, 2008.

[50] K. Sampigethaya and R. Poovendran. A survey on mix networks and their secure applications. *Proceedings of the IEEE*, 94(12):2142–2181, 2006.

[51] D. Sandler, K. Derr, and D. S. Wallach. Votebox: A tamper-evident, verifiable electronic voting system. In *Proceedings of the 17th Conference on Security Symposium*, SS'08, pages 349–364, 2008.

[52] F. Shirazi, S. Neumann, I. Ciolacu, and M. Volkamer. Robust electronic voting: Introducing robustness in civitas. In *Requirements Engineering for Electronic Voting Systems (REVOTE), 2011 International Workshop on*, pages 47–55, 2011.

[53] CACM Staff. Seven principles for secure e-voting. *Commun. ACM*, 52(2):8–9, February 2009.

[54] G. Tsoukalas, K. Papadimitriou, and P. Louridas. From helios to zeus. In *Greek Research and Education Network; Panayiotis Tsanakas, National Technical University of Athens, 2013 Usenix EVT conference*, pages 1–10, 2013.

[55] M. Volkamer and R. Grimm. Determine the resilience of evaluated internet voting systems. In *Requirements Engineering for e-Voting Systems (RE-VOTE), 2009 First International Workshop on*, pages 47–54, aug. 2009.

[56] T. Wan and W. Liao. Cryptanalysis on polynomial-based e-voting schemes. In *2010 International Conference on E-Business and E-Government (ICEE)*, pages 436–438, 2010.

[57] Wikipedia. Homomorphic encryption, 2012. [Online; accessed 21-October-2012].

[58] A. C. Yao. How to generate and exchange secrets. In *Foundations of Computer Science, 1986., 27th Annual Symposium on*, pages 162–167, Oct. 1986.

[59] S. Yun and S. Lee. The network based electronic voting scheme suitable for large scale election. In *The 6th International Conference on Advanced Communication Technology*, volume 1, pages 218–222, 2004.

[60] X. Zhao, L. Li, G. Xue, and G. Silva. Efficient anonymous message submission. In *INFOCOM*, pages 2228–2236, 2012.