Secure Email Voting

6.857 Computer and Network Security, Spring 2020

Jenning Chen Claire Hsu Karunya Sethuraman Ashwin Srinivasan

***Abstract*—With the rise of online voting systems, voting has become easier in past decades in the face of challenges at the polls, like voter accessibility, voter purges, increasing registration requirements, and voter intimidation. Though online voting systems clearly improve accessibility, they also introduce several security concerns, from voter anonymity to vote authentication. In this work, we present a secure email voting scheme, including strategies for secure registration, voting, and authentication. We focus heavily on an ideal mail scheme and propose an overlay network that utilizes onion routing to provide users with anonymity. Our email voting system also allows users to verify registration and vote information, to protect against malicious vote modifications. We evaluate our various schemes by demonstrating how they protect against leaking both registration and vote information to adversaries that can spoof users, adversaries that can spoof election authorities, and MITM attacks. Such an email voting scheme has implications in further increasing accessibility to voting, especially in times like the COVID-19 pandemic, where remote voting will be paramount in upcoming elections.**

I. Introduction

Our project focuses on email voting, or transmitting votes via attached ballot forms to emails sent to election authorities. While this scheme has been adopted in many countries worldwide in varying scales, the security implications of such online voting schemes have also been widely recognized. Most recently in the wake of the COVID-19 crisis, the EU Parliament has moved to email voting as a replacement for in-person voting [1]. Members attach signed ballots to emails for the parliament's secretariat to manually count. Similar implementations have been adopted in the United States - in conjunction with the Uniformed and Overseas Citizens Absentee Voting Act (UOCAVA), email voting has been accepted in the United States for nineteen states in special cases [2]. Thus, these schemes exist in both public and private forms of voting.

However, many researchers have pointed out the shortcomings of email voting - Prof. Halderman from the University of Michigan claimed "e-voting where the results are public is relatively low risk," [3] but in the scope of our project (which aims to tackle private voting via email), many security risks remain. Concerns like confidentiality, man-in-the-middle manipulation of votes, and malware attacks (on both individual email accounts and email servers) all pose unique challenges for email applications. For example, at the 2018 DEFCON conference, researchers were able to alter an emailed vote without detection [4]. Email servers also must follow secure routing and source and destination authentication, and with the existence of multiple email services with varying levels of security levels, verifying votes sent via email can be difficult.

II. Prior Work

A variety of electronic voting systems have been implemented over the past few decades all over the world. A notable example is the 1970s electronic voting machines implemented in the US federal government, where the member of the body would insert a unique identification card, and punch a button to indicate their vote. A similar system continues to be used today in the US House of Representatives, with the modern day incarnation featuring LCD screens and syncing with C-Span to aid in televising votes. Over the years, this innovation has allowed more business to be conducted, as these electronic votes take anywhere from 5 to 15 minutes, as opposed to a roll call vote that could take 30 minutes or more. As the Senate has fewer members, it has opted to stick with roll call votes instead of moving to electronic ones.

In the United States, there is no "centralized election authority," unlike many other nations. The Congress can enact relevant legislation, but states regulate and organize their own elections. A report, "Securing the Vote," published by the National Academies of Sciences, Engineering, and Medicine, outlines some strengths and weaknesses of current voting schemes. They recommended that in cases of voting by mail, or absentee voting, which email voting would likely fall in, the system should be designed to support voters "easily check[ing] ... whether his or her market ballot has been received and accepted." With reference to electronic systems, they recommend "backup plans," regular security assessments, and that all machines that "do not provide the capacity for independent auditing ... should be removed ... as soon as possible." However, they also recommend human readable paper ballots as the most reliable technology.

A common concept as technology comes into contact with legacy voting processes is "strongly software independent voting system," which provides a way to audit the results by preserving the audit trail and allowing for both compliance audits and risk-limiting audits to be performed after the election. The idea of auditing is also prevalent, for good reason, as it allows for a way to trust the incorporation of technology into the future of our democracy. Another common proposal is for centralized guidelines on conducting federal elections and "step-by-step procedures for conducting a risk-limiting audit.

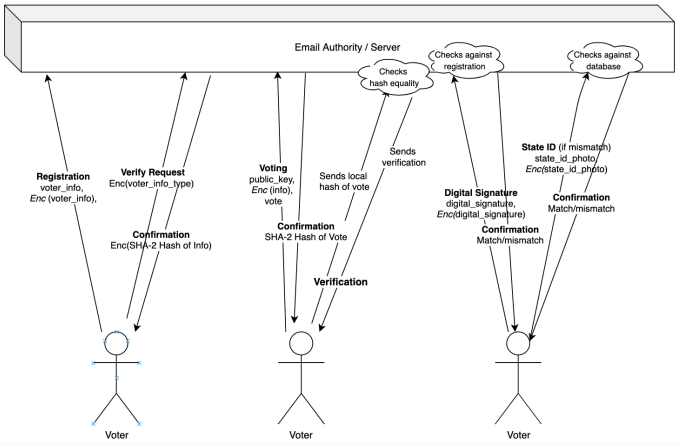

Fig. 1. Voter registration scheme (left), voting and vote verification scheme (middle), and voter authentication scheme (right).

In federal elections, states such as Georgia and Texas use electronic voting machines during elections. These machines have been called into question due to concerns that they are "flipping" votes in close races, such as Kemp v. Abrams in Georgia, or O'Rourke v. Cruz in Texas. This is a problem, as no one wants votes counted incorrectly. However, the problem is compounded when those in charge of overseeing the election process, as Secretary of State, are also running in the race; this was the case in Georgia, with Brian Kemp. With slow and unreliable machines, voters are left without trust in the system and the results of an election are disputed, clearly showing that new innovation in this area is sorely needed. Often, these old machines are found in local governments that lack the funds to upgrade, resulting in further socio-economic imbalance. In Brazil, researchers found that there is a socioeconomic disparity when electronic voting was introduced, which is also an area in which innovators must be thoughtful.

III. Design Goals

For this project, we propose a secure secret-ballot email voting system. The majority of the focus will lie in details of the voting procedure, such as secret-ballot casting, voter registration, authentication, and verification. In addition, we delineate key characteristics of the email service necessary to guarantee a secure and functional system. Although current email services trade security for convenience and generality, we propose modifications which can balance increased security with the practicality of email such that the resulting service is an appropriate conduit for secret-ballot voting. These design decisions will be justified according to related work, class material, and projected use cases of our model. We also evaluate how our proposal defends against possible attacks, such as spoofing and MITM, as well as how it mitigates its vulnerabilities.

IV. Design Overview

Our secure email voting system revolves around encrypted communication between voters and election authorities, as well as hard-to-replicate verification request/response schemes. We consider modifications to existing mail systems, accommodate our system to fit into existing voter registration systems, and propose a two-factor authentication system for voter authentication. Furthermore, our mail client introduces an onion routing scheme to ensure anonymity and confidentiality across insecure existing mail routing systems.

A. Voter Flow

Voters first register by submitting registration information via email to the email authority. This can happen using an existing user mail client (no extra accounts required). Voters can later verify their registration by sending a registration verification request containing the information they would like to verify, and will receive a hash of their requested information.

Voters similarly vote by submitting their vote via email with their public key as a unique identifier. Voters can later verify their vote, and based on the verification response, can determine whether their vote is correct.

B. Election Authority

The election authority can access user information and votes, though how the data is stored on their end is not within the scope of our project (however secure their existing voter databases are). We assume an honest election authority, so once the EA has access to votes and voter information, we assume they will not maliciously modify information or votes on their own databases. (This does not, however, follow for dishonest adversaries pretending to be EAs, which our system protects against.)

C. Assumptions

Every voter has access to email either from their own personal device (computer, smart phone, etc.) or from a publicly-accessed electronic device (library computer, for example). The purpose of our system is to make voting more accessible for people who are not able to come to a public voting site on election day by making it available electronically. Since 75.8% of Americans use email, we assume that these people have the appropriate device to access their email [5].

V. Voting System

A. Email Service (modifications)

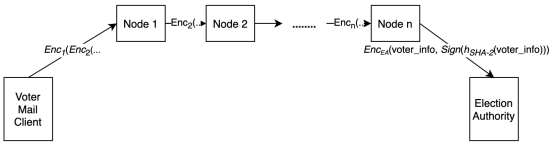

Fig. 2. Routing voter information through intermediary nodes to the EA.

1) Security Requirements: We focus first on the delivery of messages between the voter and the server (election authority). Our email service demands several facets of security in order to support an effective voting system:

- *Confidentiality*: email data sent from and received by clients should remain confidential, in line with private ballot voting
- *Anonymity*: any party besides the client should not be able to associate a client's email data with his identity, facilitating vote anonymity. With anonymity, if the receiver is compromised by an adversary, no mapping of voter identity to vote behavior can be determined.
- *Integrity*: email data cannot be changed by unauthorized users, thus precluding vote tampering/fraud and allowing for accurate tallying of ballots
- *Availability*: email servers should remain available, establishing user trust and streamlining verification and ballot recounts. By implication, servers should be resilient to DDoS attacks.

The basic function of an email service is to deliver email messages between clients and recipients. Some of today's most popular email service providers include Gmail and Outlook.com, which prevail in convenience and usability, while security is often downplayed in importance since email contents generally do not carry sensitive information. While such services provide TLS encryption, they do not automatically provide end-to-end encryption to defend against sending private data.

2) Email Client Modifications: However, in the context of voting, security becomes the most important guarantee. It is apparent that the email service must undergo modifications to achieve the four enumerated goals of confidentiality, anonymity, integrity, and availability. For our design, we draw inspiration from Tor, a network overlay which hides the identities of its users through a large circuit of relays. We thus delineate our modifications below:

- *Onion routing*: Email data is sent through a network of relays, which is owned and operated by a large number of public institutions and volunteers trusted by the election authority. The specific path of relays taken is randomly determined, deterring collusion, and the data is first encrypted repeatedly and subsequently sent through the series of relays. Each relay decrypts an encryption layer as if it is peeling back an onion layer, and each decryption reveals the next node in the path. Relays cannot tell if the previous relay was the originator, thus protecting the source node. The last relay in the path decrypts the last layer, and sends the packet to the destination node. Our onion routing scheme lies in the application layer, so considerations about packet drops or traffic handling is not addressed in this work. By implementing onion routing, we enable anonymity.
- *Introduction nodes and rendezvous points*: Through the relay network, we can also hide the public IP address of the server via introduction nodes. Such nodes are designated relays, regularly contacted by the email server to determine if any clients demand its service. The addresses of these introduction points are published with the corresponding service, so clients know which nodes to contact if they are interested in the email service. Once a client contacts an introduction node, it alerts the node of another relay, a rendezvous point, to which it will establish a connection. The introduction node appropriately communicates to the server the rendezvous point, and the server establishes a connection to the point as well. Through onion routing, the server and the client remain anonymous and hidden behind the rendezvous point, with shared knowledge of the server's public key to enable the necessary encryption. This allows for the public IP address of the server to remain private, precluding DDoS attacks and improving availability.
- *PGP*: Emails are secured through PGP end-to-end encryption, a scheme which combines both public cryptography and symmetric encryption, allowing for confidentiality. Furthermore, emails are digitally signed through PGP with help from the RSA algorithm and SHA-2 hash algorithms, ensuring not only integrity but also authenticity. This can be seen as output from the last node in the relay circuit in Figure 2.

Two options exist for how voters will register and perform voting procedures. The first is that voters use their existing email clients. The mail clients will forward the relevant mail to the voting network described above, requiring voters to manually specify the encryption option before sending, much like those offered to users of Outlook.com, Gmail, and the like. Such an option requires perfect behavior from the voter to always enable encryption, as well as cooperation from popular email services to invest in and incorporate such features into their interfaces. The second option is that voters register with a new email service that automatically utilizes our secure network. Users must log-in via a portal, providing a password which will be stored by the election authority, and their connection will be secured by HTTPS for an added layer of encryption. While such a case would require voters to create an entirely new email account in order to utilize email voting, we view this as the most secure option.

3) Tradeoffs, Shortcomings: Such a service will ensure that users' ballots will be privately and correctly received, thus upholding the essential requirements of a voting system. The main tradeoff for increased security is performance; however, such a tradeoff is tolerable as each voter is expected to send at most a few votes, and the voting period is typically large. Some possible attacks include traffic analysis, where an adversary analyzes connection records and timing patterns to determine the path of a packet, as well as exit node vulnerabilities, where once the last node decrypts the last layer of encryption, the message contents are intercepted. A traffic analysis attack can be mitigated through bundling messages together, and exit node vulnerabilities are mitigated through end-to-end encryption, which protects the last decrypted message in the relay circuit.

B. Voter Registration

Several challenges to voter registration have been identified in the past decades, from misinformation on social media to laws that disenfranchise voters by requiring specific types of identification. Incorrect voter purges and voter intimidation [6] has also risen as a barrier to voting in elections, disproportionately suppressing the vote of minority groups such as people of color and young voters. The percent of minority voters who are eligible and registered to vote is consistently 10-20% lower than that of white eligible and registered voters.

Current voter registration methods involve registering online, submitting paper forms in the mail, or registering in person at an election office. Online options, like online registration systems, help solve these accessibility problems, but must fulfill several security requirements such as secure data storage, access control, and the authentication of existing data [7]. As a result, we introduce an email-based voter registration system that fulfills these requirements, specifically maintaining confidentiality of voter registration and ensuring authenticity of election authorities.

1) Ensuring Confidentiality of Personally Identifiable Info: To ensure confidentiality of identification documents and other voter information, the voter must use a secure mail client - we propose several modifications and additions for our "ideal mail client". This client would be implemented similar to Tor, protecting the identity of voters while maintaining confidentiality of information. Utilizing an onion routing scheme would ensure multi-layered encryption of voter contact information, like name and address, and voter registration documents, like driver license numbers and photos. A problem raised in traditional email schemes is that email messages must travel through numerous routing switches, compromising security at each step. With an onion routing based system, we ensure perfect forward secrecy, removing this risk. Individual adversarial nodes in this network would fail to compromise the confidentiality of the system, preventing MITM attacks. This scheme also prevents the public IP address of the registered voter from being transmitted across a message route, hiding the identity of the sender.

2) Preventing Impersonation of Election Authorities: To ensure that voters are communicating with the actual election authority as opposed to an adversary posing as the authority, the email client must first ensure the identity of the mail recipient. As in many other proposed email voting schemes [8], in our scheme, the client must also verify the certificate of the voter registration recipient (ideally the election authority) by querying a third party, trusted certificate authority. This certificate exchange happens prior to the registration information exchange to verify the recipient's identity, as in Figure 3.

Following transmission of voter documents and information, existing voter registration databases can be utilized to increase simplicity and accommodate existing workflows.

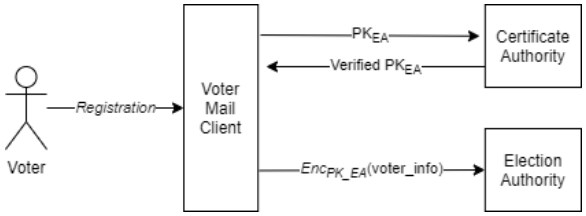

Fig. 3. Voter and Certificate Authority Exchange

3) Verification of Correct Registration: To verify that their information was received correctly, users may request certain fields of information, such as "address" or "name". The authority confirms the field by sending back a hash of the voter's requested information. This scheme ensures that only the election authority can correctly reply with the voter's requested information, but no plaintext voter information is ever transmitted. In the case of adversarial voters pretending to be other voters, no voter information can be derived from this scheme.

$$\textbf{Voter: } Enc_{PK_{EA}}(info\_type) \tag{1}$$

$$\textbf{EA: } Enc_{PK_{voter}}(h(voter\_info)) \tag{2}$$

Equations 1 and 2 detail the voter - EA registration verification exchange. Here the EA only sends back a hashed version of the voter_info, ensuring that even if the adversary were to break the encryption (e.g. by tricking the EA into encrypting with the adversary's public key or obtaining the voter's secret key), they still cannot reverse the hashed field. Only the voter with the original voter_info can discern whether the fields match, and if the voter_info is indeed correct.

C. Voter Authentication

Voter authentication is an essential feature of an email voting service. This is to ensure that voter impersonation does not occur, in which someone casts a ballot in the . This could happen, for instance, if an individual has access to a registered voter's email, such as hackers who create bots to gain access to others' emails in an attempt to cast fraudulent votes for a particular candidate. This process is also important in ensuring that voters and candidates trust the outcome of an election. The authentication process should maximize the probability of keeping out fraudulent votes, while at the same time also try to minimize the hindrances that would be inevitable in such a process, so as to not discourage voters from casting their ballot online. There are several potential ways of implementing a simple process of authentication. The first method is two-factor authentication (2FA), in which the voter would have to confirm his or her identity by entering a one-time password that is sent to their mobile device through SMS. This requires the individual to possess the mobile device of the registered voter, which provides a layer of authentication security. The

benefit of this process is that it does not require much extra effort for the voter. However, there are several disadvantages to this process:

- Transmission of SMS is not encrypted, and goes through several channels between sender and recipient, so the data can be intercepted/compromised at any of these points [9].
- Voters can fall victim to SIM Swap attacks, in which their phone number could be stolen by someone who has personal information on them
- Mobile devices can be lost or stolen. If stolen, then the individual could access the voting email as well as the SMS one-time password, rendering 2FA useless.
- Some voters may incur a fee for SMS (based on their messaging plan), so this is unfair for those individuals

Another method is by having voters digitally sign, and comparing this to the signature on their voter registration using an automated signature verification software. This is currently the most popular method for voter authentication of absentee ballots [10]. Since all voters need to provide a signature when they register, this method could be used for all voters since their registration signature would be stored on our server. The downside to this method is that subtle differences in signatures by the same individual may result in a mismatch, especially for those who are elderly, disabled, those whose signature has changed as they got older, and those who have had their name changed [11]. Additionally, there may be mismatches arising due to the fact that many voters are used to signing on paper, but now may be inexperienced in signing digitally.

A third method is by using voter identification, such as a driver's license, voter registration card, or other government-issued ID. The system will access the voter's webcam or phone camera, and the voter would need to show his or her ID to the screen, and the information on the ID card would then be scanned. The voters' information would then be verified against the state's database of ID holders. The benefits of this process are that it is relatively easy to hold an ID card to the screen, and that it is highly unlikely that another individual will have possession of a voter's ID card. However, the downside to this is that it poses a disadvantage for voters who do not possess IDs, who are disproportionately minorities and the poor, as they either cannot afford to get an ID or do not have the necessary documents to obtain an ID. For example, a study has shown that 11% of Americans and 25% of African-Americans do not have government-issued ID [12].

In order to attain the goal of maximizing authentication security while also minimizing the difficulties of the process, our system will first require voters to digitally sign. If there is a mismatch between their signature and the signature on their registration, then the system will prompt the voter to provide ID. This way, it will first provide a more convenient method of authentication that should work for any voter. In the event of a false negative (signature mismatch when the user is legitimate), however, it will require a stronger measure of authentication by the user. This method of authentication is a stronger security measure than most states have for in-person voting. However, since online voting does not occur in person, which brings the potential threat of hackers casting fraudulent votes on behalf of innocent users, instances of voter impersonation could potentially become more common in online voting over in-person voting. So, it is especially important to have an extra level of security with respect to ensuring that an individual's vote is cast only by that individual.

D. Voter Anonymity and Vote Verification

When it comes to the actual issue of voting, a voting system should be designed to engender trust in all parties involved. Voter anonymity is a critical part of establishing trust with voters, who have come to expect that their votes are secret and their identity is not tied to their vote in any public manner. Research shows that this view is a factor in people's decisions to vote as well as a way to avoid coercion and voter intimidation [13]. This does not protect their registration status, party preference, or voting history, as those are all part of the public record.

On the other side of the process, voters as well as candidates must be able to trust in the results of the vote and voting process, and this is where vote verification is essential. Vote verification means that the voter can be confident that their vote was not tampered with and was counted properly for the candidate or proposal that they voted for. While our current system has many strengths, a 2018 poll by Marist showed that 47% of those polled believe that it is likely all votes would not be counted in the upcoming November election [14].

1) Ensuring Voter Anonymity: When voters are casting votes by email, a secure email client is needed, for many reasons as well as those stated in above sections. Given that the email client is secure, the traffic coming from the voter's IP address or location should not be connected to data sent to the server at the same time. However, the issue of anonymity then arises when recording the voter and their vote on the server. To ensure the possibility of recounts, maintain a proper record, and prevent spoofing, the voter must be tied to their vote in some way. Here, we assume that the voter is sending their vote in after being authenticated by the system's scheme.

We propose a CCA secure asymmetric encryption scheme, such as RSA-OAEP, where the voter's data is encrypted and stored adjacent to their vote. To encrypt the human data pertinent to their vote so that only they can verify the vote, they must send their public key, personal information encrypted by their public key, and vote, as follows:

$$\textbf{Voter: } PK_{voter}, Enc(\text{voter\_info}, \text{vote}) \tag{3}$$

We need to record all three data values together, as in Eqn. 3. When counting votes, we can simply tally the votes for each public key, or each person that voted in the election. This way the only person that can unlock the voter information is

the person.

2) Ensuring Vote Verification: No vote selling is allowed – so no receipt of how the voter voted, even if they want one! This means, if the vote is sent by email, the email client must not preserve that email in that person's sent inbox or allow for screenshots during the process. Additionally, the voter must sign a disclosure that they are not letting anyone else look at their ballot while voting, except in cases such as a language barrier, which is already documented when people vote by mail or in person.

The other part to address is how we can allow the election officials, candidates, and the voters themselves, most importantly, to confirm and verify their own vote.

Verification Request: First the user generates a local hash of their vote (denoted as HV in Eqn. 4) along with their public key, and the verification request is encrypted with the public key of the election authority.

$$\textbf{Voter: } Enc_{PK_{EA}}(PK_{user}|HV) \tag{4}$$

The election authority receives the challenge denoted in Eqn. 4, decrypts the request, and replies with a verification response, based on whether the received vote hash matches the expected vote hash on the EA server side.

Verification Response: To indicate correctness, the EA sends the following verification response to the voter:

$$\textbf{EA: } Enc_{PK_{voter}}(\text{info\_type})|hash(info)) \tag{5}$$

In Eqn. 5, info_type is a string referring to "vote", "address", "first_name", and so on. Once the voter decrypts the response, the voter can easily see whether their information is correct - if correct, then the verification request has succeeded. If instead, the EA sends back a $(\text{info\_type}, h(info))$ pair that does not match (e.g. ("name", voter address)), the verification request has failed. We note that the info_type used is not important - rather, it serves to simply allow the user to verify the EA's response without allowing adversaries attempting to pose as the EA to intercept and send back valid verification responses.

3) Preventing Multiple Votes: To prevent a voter from voting multiple times, there will be a table stored with the EA containing the public keys of all the users who have voted so far. When the EA receives a verification request and decrypts it, it will first check whether the public key of the user is already in the table. If so, then it will reject the vote, as it has already received one from that user.

VI. Evaluation Against Attacks

A. Man in the Middle Attacks (MITM)

As described previously, the email service precludes MITM attacks through a PGP digital signature, allowing the receiver to verify the integrity of the message. Therefore, it becomes incredibly difficult for an adversary to tamper with messages. Furthermore, targeted MITM attacks are difficult due to anonymity. Onion routing provides anonymity guarantees across adversaries in intermediary nodes within email routing, and another layer of anonymity is guaranteed by a secret key and public key that the voter has, where the public key does not expose any information about the secret key or about the plaintext voter information it is encrypting. In this way, the voter cannot be targeted by a MITM attack. Transmitting voter information or vote information (as opposed to simply transmitting a hash for confirmation, which, under ideal hash function guarantees, has no way of being decoded) is limited to the necessary scope of functions.

Similar to registration, the voter authentication scheme will utilize onion routing to ensure the confidentiality of users' ID photos and signatures as well as their anonymity by hiding the IP address of the sender, so that a voter's identity cannot be tied to their vote. It will also utilize the PGP digital signature to prevent MITM attacks on this data while in transmission.

The vote verification scheme ensures that only the election authority may know who the requests are from, and see the hash of the vote. Because the election authority (and only the EA) should have access to the voter's vote and information in their local database server, the election authority 1) is the only party able to honestly evaluate the voter's hash and 2) is the only party able to send a correct verification response package. We note that only the voter can decrypt this response message to test validity. This scheme also protects against MITM attacks, given that only the election authority will be able to garner information from incoming verification requests and send back accepted verification responses (hashed voter-specific information, as opposed to a "False", "True" scheme). We also note that adversaries posing as voters cannot verify another person's vote, as they cannot decrypt the verification response.

B. Spoofing

Spoofing voter registration is a difficult problem that is not perfectly solved by current online voter registration systems, and we acknowledge that our scheme does not protect against users registering under false identities. Rather, we secure our email registration system to the level of existing online voter registration systems, showing that all information required for registration can be securely sent over our scheme. Adversaries wishing to verify someone else's registration will not be able to decode the encrypted verification response without the voter's secret key, nor interpret the hash of the voter's registration information.

The anonymity of the voter is preserved by a secret key and public key that the user has. Since the voter only votes once, it would not make sense to send fraudulent messages encrypted with the same public key, and again the secure email ensures that the public key is never exposed in transit. Therefore, it is secure against spoofing.

In the vote verification scheme, the user waits for a yes/no message from the system. While the adversary could try to

spoof this message, the tor-like system of onion routers would make it very difficult to figure what recipient to send the spoofed message to. Guessing the hash of the vote would similarly be difficult, and again the secure email client would protect the information in any email sent to/from the voter.

In the authentication scheme, the only possible way of spoofing would be to forge the voter's digital signature such that it matches exactly with when the voter signed in registration. This could only possibly be done by a family member or close friend who is familiar with the voter's signature. For the state-issued ID verification, there is no way to be authenticated unless they possess the voter's ID, which is a highly unlikely situation. However, this scheme primarily protects against hackers, which would be the largest threat to an online voting system in terms of scale.

VII. Conclusion

In the above, we showcase a design for a secure email-voting system which fulfills the necessary security requirements of confidentiality, anonymity, integrity, and availability. Namely, our email service utilizes onion routing and end-to-end encryption to ensure that ballot delivery from the client to the election authority is secure. Our scheme introduces novel methods to handle secure voter registration and registration verification while building on top of existing voter registration databases. Our service also includes an authentication scheme that takes steps to prevent fraudulent voting while at the same time not impeding the online voting process. By ensuring voters stay anonymous, we meet the standards outlined by our Constitution as well as expected by voters and the general public. By allowing verification, we make sure the voter trusts the system at large, and that candidates trust the outcomes of elections, whether wins or losses. We hope this serves as a model for a future with technology harnessed to make voting more secure and equitable.

References

[1] N. C. of Ministers and Unesda, "[coronavirus] meps vote by email on new coronavirus measures."

[2] W. Underhill.

[3] N. Lomas, "Eu parliament moves to email voting during covid-19 pandemic," Mar 2020.

[4] T. Johnson, G. Gordon, and C. Condon, "Can hackers tamper with your vote? researchers show it's possible in nearly 30 states," Aug 2018.

[5] "U.s. e-mail reach 2019," Sep 2015.

[6] R. Ayala, "Voting problems 2018." www.brennancenter.org/our-work/analysis-opinion/voting-problems-2018, 11 2018.

[7] D. Lynch and K. Brangoccio, "Securing voter registration systems," Jul 2018.

[8] G. Vinodu and M. Sebastian, "Remote internet voting: developing a secure and efficient frontend," *CSI Transactions on ICT*, vol. 1, 09 2013.

[9] D. Price, "It's time to stop using sms and 2fa apps for two-factor authentication," Feb 2018.

[10] T. Dybdahl and W. Underhill, "Verification of absentee ballots," Jan 2020.

[11] L. Carpenter, "Signature match laws disproportionately impact voters already on the margins," Nov 2018.

[12] "Voter id 101: The right to vote shouldn't come with barriers," Dec 2018.

[13] C. M. Dowling, D. Doherty, S. J. Hill, A. S. Gerber, and G. A. Huber, "The voting experience and beliefs about ballot secrecy," *PLOS ONE*, vol. 14, pp. 1–14, 01 2019.

[14] M. Parks, "Npr/marist poll: 40 percent of americans think elections aren't fair," Sep 2018.