THE FUTURE OF VOTING
End-To-End Verifiable Internet Voting

Expert Statements

10 July 2015

CONTENTS

1 JOSH BENALOH AND CONCURRING EXPERTS

RONALD L. RIVEST, PHILIP STARK, VANESSA TEAGUE, CONCURRING

THE VIABILITY OF RESPONSIBLE INTERNET VOTING

Remote voting[1] entails significant risks above and beyond those of in-person poll-site voting. Included among these are risks to integrity – as remotely-cast ballots may pass through numerous hands without independent observation – and risks to privacy – as voting takes place without the benefit of publicly-enforced voter isolation.

Internet voting substantially exacerbates the risks of remote voting by making it possible for small problems to be magnified and replicated on a large scale. Careless or malicious errors, intrusive malware, and unforeseen omissions – all of which can be caused by individuals or very small groups – can cause very large numbers of votes to be changed and the privacy of large numbers of voters to be compromised.

The technology known as end-to-end (E2E) verifiability allows individual voters to verify that their intended votes have been properly recorded and that all recorded votes have been properly counted. When applied to in-person voting, E2E-verifiability provides new assurances to voters by allowing them to check for themselves that the results of an election are correct. When applied to Internet voting, E2E-verifiability mitigates some of the risks described above – but does not eliminate them: voters are able to check that their ballots are properly recorded and counted, but malware can still compromise privacy, prevent voters from casting their ballots, and otherwise hinder voters.

Although E2E-verifiable election technologies have existed for more than thirty years, their use has thus far been limited to small demonstration systems and private elections for student governments, professional societies, and the like. E2E-verifiable elections produce new challenges and complications for implementers and administrators. They represent a new and different paradigm for elections – substantially replacing the notion of verification of election equipment with that of verification of the integrity of individual elections. As such, it is important to act deliberately and gain experience with E2E-verifiability in more manageable environments before attempting to deploy E2E-verifiable elections in their most challenging environment: the Internet.

These realities lead us to two principal conclusions.

- Public elections should not be conducted over the Internet using systems that are not end-to-end verifiable.
- End-to-end verifiable Internet voting systems should not be used before end-to-end verifiable poll-site voting systems have been widely-deployed and experience has been gained from their use.

The second of these two principles is also necessitated by the fact that an E2E-verifiable election must have a tally to verify, and if an E2E-verifiable system is used only for remote voters, then the votes of these remote voters must be separately tallied and reported. Few jurisdictions are willing to segregate and report the tallies of local and remote voters separately.

We take no position here as to whether the integrity benefits of E2E-verifiability and the privacy benefits it makes possible outweigh the risks of remotely-executed large-scale corruption of an Internet-based election, but we are agreed upon the conclusions that “naked” Internet voting is dangerous and irresponsible and that E2E-verifiability should be deployed in the less risky and more manageable scenario of in-person poll-site voting before it is deployed in the wilds of the Internet.

[1] Remote voting is defined here as voting without the benefit of the public monitoring that takes place in a traditional poll site.

2 DAVID JEFFERSON AND CONCURRING EXPERTS

CANDICE HOKE, RONALD L. RIVEST, BARBARA SIMONS, PHILIP STARK, AND VANESSA TEAGUE, CONCURRING

2.1 ELECTION SECURITY IS NATIONAL SECURITY

In a democracy election security is a key part of national security. The very legitimacy of government depends on the fact, and also the public perception, that the outcome of every election fairly represents the will of the people. We must be assured that no part of the voting process is unfairly manipulated to produce a different outcome, that all and only those who are eligible to vote have the opportunity to do so, that no one votes more than once, and that the privacy of the ballot is not compromised.

In this paper we demonstrate that while end-to-end verifiable (E2E-V) voting systems have a great deal to offer, but when they are embedded in an Internet voting context (E2E-VIV) they still do not provide sufficient security to prevent remote attackers from silently modifying votes and changing the outcome of elections undetectably, or from disrupting the election and disenfranchising a large number of voters. Fundamental security problems remain with E2E-VIV systems for which there are no practical solutions in sight and that will not be resolved in the foreseeable future.

2.2 VERIFIABILITY

Election security depends on *verifiability*. After an election is closed there must remain enough evidence for anyone who doubts the results to re-examine and rationally determine whether the winners were called correctly. We need to verify that only eligible voters voted, that no votes were lost, or duplicated, or modified, that no phony votes were inserted, that every eligible voter who tried to vote was able to do so, and that all the votes were counted correctly, once and only once. That evidence trail has to be *end to end*, spanning the entire data path through which the ballot data travels, from the voters' heads to the final result. We must be able to reconstruct the vote totals (or statistically audit them) from the original unmodified ballots or copies of those ballots that are provably identical to the originals as intended by the voter.

The traditional approaches to verifiability are all based on physically secured and indelible paper ballots (or paper cast vote records) that can be recounted or audited by humans without having to trust any software or complex machines. The goal of end-to-end verifiable (E2E-V) systems is to use cryptographic protocols to achieve for all-electronic voting systems the same (or higher) level of confidence in the election outcome for electronic voting systems as is achievable with paper-based systems. The goal of end-to-end verifiable *Internet voting* (E2E-VIV) systems is the same, but specifically extended to the much more difficult security context of remote voting from private platforms and devices over the public Internet.

Some approaches to end-to-end "verification" at first sound great, but fall far short. It is not sufficient, for example, to provide a feature whereby each voter can verify that her own ballot was correctly transmitted to the server over the Internet. First, it is difficult to provide verification capability without also making it possible for the voter to prove to a third party how she voted, thereby enabling automated vote selling and voter coercion. But even if that problem were resolved and if every voter verified that her voted ballot was properly received, it would still not be possible to demonstrate that no phony votes, unassociated with any actual voter, were inserted.

2.3 THE POWER OF E2E-V SYSTEMS

End-to-end verifiable voting systems are a major conceptual and mathematical step forward from conventional voting systems. Through advanced cryptographic techniques these varied systems, while differing in many ways in their architecture and in the roles of insiders, share the following fundamental security properties:

a) **INTEGRITY.** Once a voter successfully enters her ballot into the E2E-V system it cannot be undetectably lost or modified in any way, even in the presence of bugs or malicious logic.

b) **PRIVACY.** Once a ballot enters the E2E-V system it is encrypted, so that there is no way that the privacy of the ballot to be violated subsequently.

c) **COUNTING ACCURACY.** The ballots cannot be miscounted without that fact being detectable.

d) **UNIVERSAL PUBLIC VERIFIABILITY.** The systems output and publish sufficient verification data so that *anyone* can verify that no ballots were lost or modified and that the votes were properly counted. The verification data provides essentially a *cryptographic proof* that ballot integrity was preserved and the counts are correct. Anyone is free to run a verification program over the verification data to confirm it. You don't even have to trust the official verification program—you can use one from a source you trust, or if you have the skill you can write your own.

e) **OPENNESS AND TRANSPARENCY.** The code for E2E-V systems is generally open source. The mathematical principles underlying the E2E-V security guarantees have been vetted by many cryptographic experts and are open and public. And the specifications for proof checkers are also publically documented so that mutually suspicious political groups can hire their own experts whose independent election verification programs, if correct, must agree.

These powerful E2E-V security properties are not shared by any traditional voting system. In a precinct voting context they make E2E-V systems essentially invulnerable to ordinary software bugs, to malicious code inside (but not outside) the voting system, to transient hardware faults, and to most kinds of insider fraud (at least without a large conspiracy). Such failures will at least be detected because any lost or modified ballots and any miscounts will be flagged whenever anyone runs a verification program. This is in strong contrast to other forms of purely electronic voting systems which are unverifiable, and in which bugs or malicious logic can cause errors that are totally undetectable. The wrong people may wind up taking office without anyone knowing that the election results were incorrect.

For these reasons it is appropriate to consider E2E-V systems in any electronic voting situation. E2E-V adds truly powerful security guarantees, particularly in a precinct voting situation where voter authentication is done in person and where we have good reason to presume that the certified software in the voting machines is not malicious.

But as we explain in the next section, these E2E-V security guarantees *do not fully extend to Internet voting systems*. E2E-V Internet voting systems (E2E-VIV) have exploitable security holes for which there are no good solutions today and that preclude them from being suitable for use in public elections for the foreseeable future.

2.4 REMAINING UNSOLVED SECURITY ISSUES WITH E2E-VIV SYSTEMS

Once the voters' choices are safely input to a precinct-based E2E-V system many, but not all, of the security guarantees described in the last section follow directly. But that is a key qualification: these guarantees begin *once the voters' choices are safely input to the E2E-V system*, but not before. Unfortunately, when an E2E-V system is embedded into the Internet voting context as an E2E-VIV system, new security problems appear that the E2E-V guarantees do not address and that cannot, with any current technology, be fixed. The problems with E2E-VIV systems arise *before the votes even enter the system*. In this section we enumerate the remaining difficult security problems that will have to be solved definitively before we can consider deploying an E2E-VIV system.

VOTER AUTHENTICATION

A central issue with all remote, online voting systems is voter authentication. The voting system must be able to positively identify the voter in strong a way, so that it is essentially impossible to avoid the authentication process, and impossible to fool it so that an ineligible person is allowed to vote or that someone can fraudulently impersonate another voter.

Voter authentication is not part of an E2E-VIV system, but is a separate security issue. Unfortunately it is a very difficult and complex problem that remains unresolved (in the U.S. at least) for the foreseeable future.

Strong voter authentication is required for several reasons. Any online voting system must:

- verify that potential voters are duly registered or eligible to vote in the jurisdiction they attempt to vote in;
- prevent anyone from voting more than once; and
- resist vote selling, vote coercion, and proxy voting insofar as possible in a remote voting situation.

In an online voting system it is not sufficient to use the kinds of authentication commonly used in ecommerce situations, e.g. passwords, challenge-response systems based on personal information, or email-confirmations. These very weak authentication systems are more or less sufficient in commercial situations where secrecy is not so important, and where fraudulent transactions can be detected eventually and frequently be reversed or at least can be absorbed as a cost of doing business.

But in the national security context of an online election such weak authentication mechanisms will not suffice. If an attacker has the technical means to impersonate one voter, he can generally automate and amplify his methods to impersonate thousands of voters with very little additional effort. Almost every month we hear of huge data breaches at commercial or government institutions that have already allowed vast amounts of personal information on tens of millions of people to fall into the hands of criminals or foreign powers. Thus, any authentication mechanisms based on merely presenting personal information (name, address, account number, driver's license or social security number, mother's maiden name, etc.) is hopelessly compromised already, and way too weak for use in an election. Unfortunately some states have implemented such embarrassingly weak online authentication systems and have been forced to strengthen them, though they are still not sufficiently strong.

The traditional voter authentication method is based on wet ink signature matching. The voter is required, either in person at the precinct or on the envelope of a mail-in ballot, to duplicate with a new ink signature the old signature image on file from the time she registered to vote. In some states this is augmented with VoterID requirements at the polls. But there is no way to securely (unforgeably) input a wet ink signature image to a computer or mobile device and transmit it over the Internet for authentication with the ballot. Nor is there yet a way to securely and unforgeably transmit any of the usual VoterID documents.

Many people have suggested voter authentication systems based on biometrics such as fingerprints or retinal scans. For very good reasons too numerous to fully explain here none of these mechanisms is suitable for online voting. And it is important to note that while some mobile phones and tablets have fingerprint authentication devices built in, such systems authenticate the user to the device only. They do not authenticate the user to any remote service over the Internet, nor can they easily be extended to do so securely.

Other stronger, more technical authentication methods could be considered. Voters could be issued cryptographic ID cards such as the CAC cards issued to DoD personnel or like the national ID card of Estonia. Cryptographic ID cards would in principle enable voter authentication from any Internet-connected computer or device that could read them. But no U.S. state issues such IDs to its citizens or voters, and it seems unlikely that any will do so in the foreseeable future. Even if the security climate changes and people are willing to accept such an ID system, the startup and maintenance costs will be very high. Voters would have to buy computers or devices that could read the cards, and they would almost certainly have to be useful for other online purposes besides just voting in order to justify the costs involved to both the government and the voter.

The fact is that the U.S. has no strong, universally deployed online citizen identification and authentication system, and none is on the horizon. While strong remote voter authentication is not a fundamentally unsolvable problem, it is an immense practical problem that has to be dealt with before we can consider deploying any online voting system, including E2E-VIV systems.

CLIENT SIDE MALWARE

In an E2E-VIV system voters compose and input their vote choices on a privately owned (hence unsecured) platform, either a PC or mobile device. If the voting platform is infected with malware or spyware, it is complicated to prevent the votes from being modified or reported to a third party, and impossible to prevent them from just being thrown away by the malware before the ballot is encrypted and enters the E2E-VIV system.

The malware threat is ubiquitous now and is fundamental to all online voting systems. No device is safe from malware infection. No software safeguards such as commercial antivirus systems are very effective against it. There are hundreds of ways that malware can infect a voting platform, sometimes by sophisticated technical means and sometimes by tricking users into taking unsafe actions. There are hundreds of places in the huge multilayered software ecosystem of a PC or mobile device where malware can hide and be launched from. And there are thousands of hardware, OS, browser, combinations, and countless configuration choices—way too many for any possible comprehensive defense against malware. A modern PC or mobile device can easily contain software elements from a hundred different companies or open source development groups, any one of which may either be malicious or contain critical vulnerabilities that enables malicious code. Such vulnerabilities are so numerous that more are discovered all the time and vendors release security updates on a regular basis to plug the more recently discovered holes. E2E-VIV systems have no ability to prevent or even detect the actions of malware before the votes are safely submitted into the E2E-VIV software.

Malware in voters' computers can undermine the election in three fundamental ways.

i) **MALWARE MODIFICATION OF VOTES.** Malware may actually modify the voter's choices surreptitiously, before they are submitted to the E2E-VIV system. The techniques for accomplishing this without tipping off the voter will depend on the detailed architecture of the voting system, e.g. whether the client side is packaged as a full-blown application, or as a mobile app, or a Javascript script, or a browser plugin, or some other form. But in all cases the voter's choices must be input to the PC or mobile device in the clear, unencrypted, and be processed by a large amount of system software and application/browser/script software *before* it is encrypted and submitted to the E2E-VIV system. Regardless of the E2E-VIV architecture, with today's software tools it is reasonably straightforward for malicious code to modify votes undetectably before they are encrypted.

   Depending on the design of the E2E-VIV system, it may be possible for some voters to discover after the fact that different votes were recorded for them than the ones they thought they cast. But even so, there will generally be no way to prove to election officials that the voter did not cast the votes recorded for her by mistake, or cast them deliberately and then change her mind. Whether there is a remedy for voters who claim their votes are modified by malware and falsely recorded is an unresolved question.

   We cannot generally eradicate the threat of malware. But in the special case of online voting there are techniques that in theory can prevent malware from surreptitiously modifying votes. Unfortunately they all involve additional burdens on the voter in the form of code voting, or special hardware devices independent of the PC, or a second independent communication channel to the election server that does not use the Internet, or at least is guaranteed to use an independent path from the one the votes travel. All known methods of working around client side malware involve some complication in the voting process that will be enough of a barrier, at least for the time being, to discourage many voters.

ii) **MALWARE VOTE PRIVACY VIOLATION.** Malware on a voter's computer or mobile device could allow her completed ballot to enter the E2EVIV system unmodified, but prior to that it could *also* send a copy of her votes to a third party. Unless a voter has considerable expertise and has special instrumentation running during the voting transaction there is no way for her to know it. And if the instrumentation was not in place before voting there is no after-the-fact test that can determine whether this happened, and certainly no way to reverse the privacy violation. If the voter is voting from a mobile device, as opposed to a PC, often no such instrumentation even exists today.

   In any remote voting situation there is always the possibility that someone can physically look over a voter's shoulder and watch her vote. That is a risk we live with also with paper mail-in ballots. But the main concern is not with individual cases of privacy violation, but with widespread automated spying on many online votes.

Widespread vote privacy violation can undermine democracy in two major ways. First, in situations where some people have power over others (e.g. employers, commanding officers, union supervisors, parents, nursing home management) revealing who cast which ballot can be the basis for coercion or retaliation. This may not (yet) be a widespread concern in the U.S., but it certainly is in other countries.

Also, automated privacy violation can enable large scale, automated vote buying and selling. It is easy to imagine a scheme in which many voters are induced to sell their voting credentials, or to run a particular program while voting for a particular candidate in exchange for PayPal dollars or some other crypto currency such as Bitcoin that can be transmitted entirely online. The vote buying transaction would likely be totally undetectable by authorities. Even if the scheme eventually comes to the attention of authorities, the buyers may be long gone, or may be on foreign soil out of reach of U.S. law. In any case there would be no way to know how many votes were sold or who the sellers are. Technical tricks, such as allowing voters to vote multiple times online with only the last cast vote actually counting, are not effective when the attacker knows how the system works or when the voter cooperates in a vote sale.

As with malware that intends to modify ballots surreptitiously, there are workarounds that can prevent malware from surreptitiously revealing how someone votes to a third party. But again, they complicate the process of online voting sufficiently to be a barrier that will discourage many voters from voting

iii) **MALWARE DENIAL OF SERVICE.** The easiest and most intractable malware attack is one that simply prevents the voter from successfully voting. That can be done in many ways. The malware could make it appear that there was an error of some kind, which might be frustrating but hardly surprising to voters who might either blame themselves or their own flaky computers or attribute it to just another buggy online service. Alternatively, the malware might perfectly mimic a completed voting transaction, so that the voter believes she has successfully voted, whereas the malware would simply throw the ballot away.

Such a denial of service attack might not be very politically effective if it is applied to a random set of voters. But if it can be applied *selectively* to voters who would be likely to vote in a way that the attacker does not like, then it becomes a powerful partisan fraud tool. The malware writer may want to make a good guess as to how the voter will likely vote before deciding whether of not to block her from voting. Fortunately, there are many clues in a voter's computer or mobile device, such as browser history, to indicate at least a likely party preference or social class, and that is probably all the information the malware would need.

Some voters may be sophisticated enough to detect that their ballots were never included in the count, especially during the post-election verification stage when some might discover that there is no record that they voted. But any particular voter would find it almost impossible to prove to election officials that she actually tried to vote online but that malware prevented it. Perhaps she simply never really tried to vote, or for some other technical reason not related to malware she had been prevented from successfully voting. There would be no evidence anywhere accessible to officials that could help them diagnose the situation. Even if the voter brought her computer in for forensic examination by experts, chances are that the malware would have erased evidence and erased itself, leaving no trace.

And finally, even if an obvious widespread malware denial of service attack were somehow discovered, there would be no way to estimate how many ballots were lost and how many voters were disenfranchised. The E2E-VIV system does nothing to help with such an estimate because the ballots are discarded by the malware before they ever enter the E2E-VIV system.

Unfortunately, while there are (at best inconvenient) workarounds for malware that aims to surreptitiously modify a ballot or send it to a third party, *there is fundamentally no workaround for malware designed to just prevent voting*. Well-designed malware would make the voter believe she had successfully voted, and she might never discover until it was too late that she did not. Even if she did discover it, the only recourse would be to vote from a different, uninfected PC or device, *but she almost certainly would not know that malware was the cause of the problem and would likely not know to vote from a different machine!*

Malware is a profound, absolutely fundamental problem that has been with us since the dawn of the PC age or before and will be with us for as far into the future as we can see. There is fundamentally no way to totally eradicate client side malware, or totally immunize against it, or even detect its presence. Malware is getting ever easier to write because templates, kits, libraries, and exemplars of successful malware are widely available to aid attackers, and because the payoff far exceeds the risks of getting caught. It is estimated that anywhere from 10 to 30 percent of all PCs in the world are infected with malware, and even more when spyware is included. And there are probably no reliable estimates as to the fraction of mobile devices similarly infected.

Finally, even if an easy to use, accessible workaround for client side malware is invented that preserves vote integrity and privacy, it will still not be possible to prevent a malware denial of service attack that just blocks voting. Such denial of service attacks are in a theoretically different category and there will never be a general way to thwart them all, nor a general way for voters or election officials to unambiguously recognize one. Even if it is recognized, there will be no way to estimate how many voters were affected.

The conclusion therefore, is that client side malware remains a fundamental threat that E2E-VIV systems cannot fully defend against.

NETWORK ATTACKS AND DISTRIBUTED DENIAL OF SERVICE (DDOS) ATTACKS

E2E-VIV systems are client-server protocols that execute on top of several layers of other software, including the operating system and browser on the client side, the operating system and server complex on the server side, and the various levels of TCP/IP protocol stack all through the Internet, as well as routing protocols, DNS, NTP, DHCP, and also numerous other protocols used in wireless or mobile devices. The E2E-VIV system cannot work properly unless all of this other software works properly also. We have already discussed the problem of malicious logic on the client side. But E2E-VIV software is also attackable from the server side or from the Internet infrastructure software that the E2E-VIV software depends on.

There are many ways to attack an E2E-VIV election by maliciously modifying or configuring the software in the Internet. Such attacks are called *network attacks*. Any IT person who controls a router, DNS server, or another element of Internet infrastructure is in a position to prevent votes from getting to their destinations. On the positive side, E2E-VIV systems are partly robust against such network attacks in that they cannot result in votes being falsely injected or modified without detection. This is a clear advantage that E2E-VIV systems have over other Internet voting systems. However, there is no way to prevent a network attack from disrupting the E2E-VIV protocols in a way that causes ballots to be lost, i.e. undelivered. While this will also be detectable, the malicious loss of votes in transit cannot be prevented by an E2E-VIV protocol, and it may not be possible even to estimate the number of votes affected.

One especially dangerous form of network attack is the *distributed denial of service* (DDoS) attack. In this attack, an attacker floods the server (or some other subsystem) with so much traffic or other work that it either crashes the system or else slows it to such a crawl that it is effectively down. Voters would experience a DDoS attack on a vote server as either *extreeeeemly* long waits between steps in the voting process, or total nonresponsiveness of the server, or some other error. The net effect is that large numbers of voters would simply be disenfranchised. The attack can be directed pointedly at the server side, in which case all online voters would be affected, or it could be selectively directed at certain parts of the Internet infrastructure that would affect only a subset of the voters.

We have to be able to defend online elections against DDoS attacks for two key reasons. First, they are about the easiest of all network attacks to perpetrate. There are many different kinds of DDoS attack against different parts of Internet infrastructure and different levels of software, and there are many kits available on the dark net to allow anyone from anywhere in the world to perpetrate a DDoS attack against almost any target. In fact, the means of DDoS attack are so routinized and ubiquitous that there are illegal businesses online that will conduct an attack to your specifications against any target you choose for a moderate price. You can ask for, say, a 50 gigabit per second attack for the last 4 hours on Election Day against the IP address of the (hypothetical) Cook County vote server. That would probably prevent anyone from voting online in that jurisdiction during those hours. All of those voters would be disenfranchised, but none of them would be able to prove that they were among of the victims, and election officials would not even be able to make a decent estimate of the number of ballots lost.

DDoS attacks have actually been used in real public elections around the world at least four separate times that have been made public. (Arizona Democratic Primary, 2000; Ontario NDP, 2003; Hong Kong people's election, 2012; NDP of Canada 2012). While there are various tools that can be used to *ameliorate* some DDoS attacks, there is no *general solution*, and the DDoS problem is so fundamental that there will probably never be one with the current architecture of the Internet. Vulnerability to DDoS attacks is effectively built in to its design. Hence, all E2E-VIV systems are vulnerable to network attacks that can result in disenfranchising a large number voters with no way of even measuring how many were affected. There is no fundamental defense against DDoS attacks.

2.5 CONCLUSION

E2E-V offers a dramatic improvement in the security of voting systems. While it is necessary for *any* online voting system for public elections, it is by no means sufficient. Once it is embedded in a larger *Internet voting* context fundamental new security vulnerabilities appear for which there are no solutions today, and no prospect of solutions in the foreseeable future. These include vulnerability to authentication attacks, client side malware attacks, and DDoS attacks that can be perpetrated by anyone in the world. Unless and until those additional security problems are satisfactorily and simultaneously solved—and they may never be—we must not consider any Internet voting system for use in public elections.