Going from Bad to Worse: From Internet Voting to Blockchain Voting

Sunoo Park
MIT & Harvard*

Michael Specter
MIT†

Neha Narula
MIT‡

Ronald L. Rivest
MIT§

January 15, 2020 (DRAFT)

Abstract

The public is worried about election security — understandably, and perhaps more today than in recent memory. The news teems with reports of possible election interference by foreign powers, of unauthorized voting on the one hand and voter disenfranchisement on the other, and of technological failures calling into question the integrity of elections in the U.S. and elsewhere in the world.

Some have advocated "voting over the Internet" or "voting on the blockchain" as promising ways to increase election security. This paper examines such claims, and finds them both wanting and misleading. Even taking into account the many imperfections of election systems in use today, Internet- and blockchain-based voting would drastically increase the potential for catastrophic, undetectable, nation-scale election failures.

The intuitive appeal of online voting arises partly from the perceived convenience and accessibility of voting from a computer or smartphone. But studies have been inconclusive, showing that online voting may have little to no effect on turnout in practice, and sometimes even increase disenfranchisement.[1] More importantly: any increased turnout associated with Internet- or blockchain-based voting would come at the pyrrhic cost of losing any credible assurance that votes have been counted as voters cast them, as opposed to undetectably altered or discarded. This is because electronic-only voting systems — including blockchain-based systems — will be highly vulnerable to catastrophic failures for the foreseeable future, given the state of the art in computer security.

The bulk of this article's analysis systematizes prior research about the security risks of online and electronic voting, and explains that these critiques apply equally to blockchain-based voting system proposals. The article also observes that blockchains may actually introduce new problems to voting systems, and provides a list of questions intended as a reference for critically assessing security risks of any new voting system proposals.

1 Introduction

Over the years and decades, the Internet and related technologies have demonstrated their vast potential to streamline complex processes: improving efficiency, reliability, scalability, and convenience. Many important aspects of our everyday lives are increasingly conducted online.

So it is not surprising that the question has come up: *why don't we vote online?* Thinking of the convenience of voting in a few taps on a phone, without long lines, without breaking one's daily routine or taking off from work — it seems a tantalizing prospect at first glance. But there is a fatal flaw.

Any online vote-casting system is vulnerable to catastrophic failures from vastly larger-scale, harder-to-detect, and easier-to-execute attacks than would be possible against paper-ballot-based alternatives. What's more, given the state of the art in computer security, and given the high stakes involved in political elections, online vote-casting sys-

*Researcher, MIT Media Lab, Digital Currency Initiative; J.D. Candidate, Harvard Law School; and Affiliate, Berkman Klein Center for Internet and Society at Harvard University.

†Ph.D. Candidate, MIT CSAIL (Computer Science and Artificial Intelligence Laboratory) and MIT IPRI (Internet Policy Research Initiative).

‡Director of Digital Currency Initiative, MIT Media Lab.

§Institute Professor, MIT CSAIL (Computer Science and Artificial Intelligence Laboratory).

[1] See, e.g., [28, 58, 63] and more discussion in Section 1.

tems will suffer from such vulnerabilities for the foreseeable future.

This is not to minimize the importance of convenience in elections. On the contrary, convenience minimizes barriers to voting, and is an essential goal of democratic elections — just as security is. These two goals must be balanced and optimized together, subject to realistic practical constraints.

Exposing our election systems to unprecedented catastrophic failures is no price to pay for the convenience of voting from our phones. In other words, *what good is it to vote conveniently on your phone having no assurance that your vote will be counted?* Those in favor of increasing voter turnout, reducing voter fraud, or combating voter disenfranchisement should oppose online voting because catastrophic failures undermine each of those goals. Increased turnout is only meaningful in a system that assures that all the votes are counted; and the potential for larger-scale, harder-to-detect attacks against online voting systems means increased potential for undetected fraud, coercion, and sophisticated vote tampering targeting specific subgroups of voters.

What's more, online voting may not increase turnout. Studies on online voting's impact on voter turnout have ranged from finding no impact on turnout (e.g., Switzerland [28]) to finding that online voting slightly decreases turnout (e.g., Belgium [17]) to finding that online voting slightly increases turnout but is nonetheless "unlikely to solve the low turnout crisis" (e.g., Canada [30]).[2] Studies of Estonian elections have also suggested that turnout changes due to online voting may favor higher-income and higher-education demographics [58].

Yet proposals for online voting have been gaining traction recently, in the U.S. and abroad. These proposals appear to be frequently misperceived to promote exactly those goals listed above: increasing turnout, reducing fraud, or combating disenfranchisement and coercion. A number of recent online voting proposals have been accompanied by promises of added security based on blockchain technology (e.g., [25, 68, 69]), and have continued expanding despite growing opposition and warnings from computer security and blockchain experts (e.g., [38, 39]) and some tech reporters (e.g., [7, 32]).

The most prominent example has been a mobile voting app called Voatz, which markets itself as blockchain-based. Voatz was deployed in 2018 in West Virginia for overseas military voters [71, 72] in the national midterm elections, as well as in several other U.S. states for smaller-scale (e.g., municipal or county [43, 57]) elections. Alarmingly, not only is Voatz subject to catastrophic failures as any online voting system must be, but it has persistently concealed how its system works to provide the guarantees it claims. This lack of transparency goes against basic principles for vetting security-critical systems [32, 39]. Allegations of attempted hacking of that West Virginia election have since led to an FBI investigation [16].

Another recent instance of blockchain voting involved the city of Moscow in Russia, which adopted a different blockchain-based system for its September 2019 city council elections [49]. Though they took the laudable step of publishing the system's code [67], and also invited security researchers to audit it to a limited extent [41, 48], the system was quickly shown to be gravely vulnerable — not once, but twice, the second time after a proposed fix by the city [27, 29]. While the city was receptive to the first reported vulnerability, it appears to have largely ignored the second. Yet other smaller-scale blockchain voting experiments have taken place in Japan and Switzerland [8, 65].

The recent interest in online and blockchain voting proposals appears related to a growing political enthusiasm for improving and modernizing election systems — and for increasing their security from malicious interference (a topic of particular prominence in American politics). This is a promising trend, given that historically, many election authorities have been heavily constrained by limited funding for election equipment. We hope that this enthusiasm may lead to support and adoption of more secure, more transparent election equipment (addressing the many security flaws that have been documented in existing voting systems, as extensively documented for U.S. voting equipment, e.g., in [10, 11, 12]). However, the political expediency of adopting a "high-tech" solution also poses the risk that proposals may be too quickly pursued, before allocating sufficient time and funding for inde-

[2] See [63] for a concise overview of relevant studies up to 2018, including additional references.

pendent audits and feedback from security experts. New technologies should be approached with particular caution when a mistake could undermine the democratic process. After all, election systems have been designated as national critical infrastructure implicating a "vital national interest" [34].

The surprising power of paper A natural but mistaken inclination is to *entirely replace* existing voting methods with the latest digital technologies. Some are asking: why wait in lines at physical polling places to cast votes on old, clunky voting machines, when votes could be cast from voters' modern computers and phones via the Internet — protected, perhaps, by the same security protocols used for online shopping or banking or cryptocurrency technologies?

But getting rid of paper ballots as well as outdated voting equipment amounts to throwing out the baby with the bathwater, and would make elections much less secure than the imperfect system we have today. Online shopping and banking face a vastly different types of threats from election systems, in two key ways. First, online shopping and banking have a much higher tolerance for failure — and *they do indeed fail*. Credit card fraud happens, identity theft happens [66], and sensitive personal data can be breached at massive scale (such as in the 2017 Equifax breach [19]). Online shopping and banking are designed to tolerate this kind of failure, with the banks and merchants absorbing the risks because it is in their economic interest to promote business, with insurance companies offering insurance against this type of risk, and with the government offering legal recourse for victims in certain cases (e.g., the Equifax settlement [20]). But there is no such thing as insurance against a failure of democracy, and there is no entity that can hand voters a blank slate (or monetary "compensation") after a compromised election as a bank could hand a customer a new credit card.

Theft of Bitcoin and other cryptocurrencies happens too, and has resulted in losses up to hundreds of millions of dollars [59]. Blockchain-based systems generally have fewer insurers and risk-absorption mechanisms than the traditional banking system, so losses often fall directly on those stolen from, who have no third party to whom to appeal for relief.

The second key way in which the threat profile of online banking, shopping, and Bitcoin differs from that of elections is the skill level and aims of the potential adversary. Elections can be much higher-value targets for sophisticated (nation-state) attackers, whose goal is not to make fraudulent financial transactions but to change, or undermine confidence in, an election outcome. Depending on a voter's location and likelihood of tipping an election, a poor, technologically unsophisticated voter may be the target of attacks by some of the most sophisticated operators in the world.

Thus, from a computer security perspective, securing an online voting system against attack is a starkly different — and much harder — problem from securing against fraud and theft in online shopping, banking, or other seemingly sensitive online activities. Paper ballots, perhaps surprisingly, provide indispensable protection against malfunctions or attacks on the higher-tech components of the system (as explained in more detail in Section 2).

Software independence In fact, paper ballots are the only known way to achieve *software-independence* in voting systems [51, 52]: that is, the property that an undetected change or error in a system's software cannot cause an undetected change in the election outcome.

We view software independence as an *essential* requirement for any voting system in a consequential political election. Democracy — and the consent of the governed — cannot be contingent on whether some piece of technology is behaving correctly.

Categories of voting systems This article distinguishes between four main categories of voting systems, determined by two key system attributes, as shown in Table 1. First, are votes cast *in person* at a polling site, or *remotely*? Secondly, does the system have *voter-verifiable paper ballots* or are ballots represented in a format that is not voter-verifiable (e.g., purely electronic data)?

"Voter-verifiable" means that voters must be able

[3] "DRE" stands for "direct-recording electronic." This includes any machine where votes are recorded entirely electronically. For example, many touchscreen voting interfaces are DRE machines.

Table 1: **Four categories of voting systems.** The top row (green) is *software-independent* and far less vulnerable to catastrophic failure than the bottom row (red). The bottom row is highly vulnerable and thus unsuitable for use in consequential political elections, as explained further in Section 2.

	In person	**Remote**
Voter-verifiable paper ballots	*Precinct voting*	*Mail-in ballots*
Unverifiable or electronic ballots	*DRE[3] voting machines*	*Internet/mobile/blockchain voting*

to verify *directly* (i.e., without relying on a computer) that their ballot accurately represents their intended vote. For example, a paper ballot is not voter-verifiable if the voter can never inspect it.

(There is a valid argument made that being voter-verifiable *in principle* isn't sufficient if voters don't verify their ballots *in practice* [5, 62]. We skip discussion of this significant point here, as it doesn't seem germane to our main topic.)

Not every voting system that involves the use of a mobile phone, the Internet, or blockchain technology need necessarily fall in the bottom-right category. For instance, an in-person paper-ballot-based voting system could use such technology as an auxiliary tool: e.g., allowing voters to use their phones to better understand the instructions or streamline the creation of a paper ballot,[4] and/or saving a copy of the vote cast by paper ballot in an electronic format via the Internet (perhaps on a blockchain). To be clear, this article does not oppose the use of technology in the context of in-person voting systems with voter-verifiable paper ballots.

However, all of the prominent proposals billed as "Internet voting," "mobile voting," or "blockchain voting" appear to involve remote voting with electronic-only recording of votes transmitted via the Internet: such schemes all fall in the bottom-right category. Accordingly, this article uses the terms "Internet voting" and "blockchain voting" to refer to Internet- and blockchain-based schemes in the bottom-right category only.[5] We consider "blockchain voting" to be a subcategory of "Internet voting," since the vast majority of blockchain implementations (including "blockchain voting" proposals to date) involve transmission of information over the Internet.

As the rest of the article will discuss, the top row and left column of Table 1 are strongly preferable to the bottom row and right column in terms of security risk. We consider the top row to be suitable for consequential political elections, with in-person voting being preferable to mail-in ballots wherever practically feasible (as indicated by their graduated green color). Importantly, the top row satisfies *software independence* (defined above), whereas the bottom row does not.

We consider the bottom row *unsuitable for consequential political elections for the foreseeable future*, due to its lack of software independence and the vastly greater risk of compromise as compared to the available alternatives in the top row. The reasons for this heightened risk are elaborated in Sections 2–3.

The left column of Table 1 is preferable to the right column, because remote voting systems present unique opportunities for coercion and vote selling. No remote voting system can ensure voter seclusion as can a physical polling place; and without seclusion, a coercer or vote buyer can simply look over the shoulder of a voter and confirm that she voted as she was "supposed" to.[6] In contrast,

[4] For example, Los Angeles County has allowed voters to preload decisions on their phones and easily transfer the saved choices to ballots at the physical polling place [23].

[5] This article does not need to distinguish between "mobile voting" and "Internet voting" more generally; mobile voting generally involves transmission of information via the Internet, so is a subcategory of Internet voting. This article avoids the term "mobile voting" in the interest of generality.

[6] Some proposals have been made to mitigate this problem, which may be helpful but still leave many types of attacks unaddressed: for example, allowing voters to submit multiple votes and only counting the last one as real. This is ineffective in situations where a coercer or vote buyer can

with the seclusion of physical polling sites, a coercer or vote buyer cannot be sure, rendering coercion and vote buying generally ineffective.

A number of recent pieces of proposed legislation in the U.S. have recognized the need for paper-ballot-based voting systems (i.e., the top row of Table 1) and put forward the requirement of paper ballots (e.g., [36, 54, 75]). For example, the SAFE Act [36] requires: durable paper ballots; that voters be able to inspect marked ballots before casting; that voters with disabilities have an equivalent opportunity to vote (including privacy and independence) to other voters; that voting technology be manufactured domestically; and other basic security requirements such as air-gapping.[7] However, such legislation is not necessarily likely to pass in the near future; in order to become law, it must also pass an eventual vote in the Senate.

Scope and terminology This article uses "online voting" and "Internet voting" synonymously, and in accordance with popular usage, to refer to any system where voters cast their votes via the Internet — including blockchain-based and mobile voting systems. Further, we use "electronic voting" to refer to any system where votes are cast purely electronically (i.e., the bottom row of Table 1).[8] Thus, online voting is a subcategory of electronic voting. Much of the reasoning in this article applies to all electronic voting, while some parts apply specifically to online or blockchain voting.

This article focuses on systems for vote casting and tallying (namely, the subject of the recently circulating online and blockchain-based voting proposals). Internet- or blockchain-based technologies may or may not be helpful for other parts of election infrastructure (such as election auditing or voter registration); this article does not explore such possibilities in detail. New technologies in other areas of election infrastructure should, of course, be examined carefully before deployment as well.

Finally, this article focuses on the heightened security required, and particular threats faced, by consequential political elections. Other applications calling for "voting" in a broader sense may have specific security requirements that are less stringent than those of political elections (e.g., school or professional society elections). Whether electronic voting is suitable for such applications will depend on the specific circumstances, and is not addressed in this paper. The term "election" may be read as "political election" for the rest of this paper.

Election security premises This article posits a few basic premises, listed next, and explains how catastrophic failures in online voting systems would undermine these basic requirements of a trustworthy election.

1. Election equipment may fail. The system must be designed not only to prevent failures, but also to ensure timely detection of failures when they occur: the public has a right to know about failures in the election process.
2. The election process must produce convincing evidence that the outcome is fair and accurate: that all eligible votes were cast as intended, collected as cast, and counted as collected.
3. The election system must support the right to a *secret ballot*. Secrecy of ballots is essential to protect voters from coercion and vote buying.

These premises underscore the need for software-independent voting systems. With the complexity of modern technology, and with new security vulnerabilities and breaches popping up every day, an election's integrity can never be credibly assured by a claim of the form: *"These are the election results. We believe they are correct, because the computer says so."*

Organization Section 2 defines catastrophic failures, and explains why online voting systems are highly vulnerable to such failures. Section 3 discusses blockchains and how they might be used in election systems, noting that blockchains do not mitigate any of the inherent weaknesses of online voting systems (described in Section 2), and may

monitor the voter from the time of voting until polls close (e.g., because there is not much time left until polls close, or because they live in the same home).

[7] Air-gapping means maintaining a device disconnected from the Internet and from any internet-connected device.

[8] This can include systems that use paper somewhere in the process: e.g., if votes are cast and stored electronically, but copies of each electronic vote are printed out at some point during the process after the respective voter has cast her vote.

sometimes introduce yet additional weaknesses into an election system. Section 4 provides a framework for election officials and citizens to critically evaluate voting technology proposals taking into account the state of the art in computer security. Section 5 discusses other related work. Finally, Section 6 gives our conclusions.

2 Vulnerabilities of electronic voting systems

In this section, we argue that there is a class of security flaws that so gravely undermine election integrity — and thereby, democratic legitimacy — as to outweigh any countervailing interests, and that electronic voting is far more vulnerable to such failures than paper-based alternatives.

We call these incidents *catastrophic failures*: Situations in which the election result's winning measures have been altered such that the change is undetectable and/or irreparable without calling for an entirely new election — whether due to simple errors or adversarial attack.

Importantly, even the very fact, and public perception, that the system is vulnerable to such issues may be enough to reduce an elected official's legitimacy and therefore destabilize a democracy. In other words, vulnerability to catastrophic failures definitionally undermines government legitimacy, whether or not the vulnerability was exploited by an attacker.

Even simple, well-understood tools like paper ballots are not totally immune to catastrophic failures. For instance, if an election authority is allowed to handle ballots in secret, malicious actors could undetectably destroy ballots cast against their favored candidate. If the malicious authority is crafty enough, and the margin of victory small enough, it can discard ballots such that the public may never know that any incident has occurred. This is why most election authorities leverage a number of transparency tactics that allow independent observers — including representatives from either party — to monitor and contest any part of the election process as it is happening, a feature that is greatly helped by the presence of an auditable paper trail.

Unfortunately, there is a limit to the ability of independent observers and monitors to prevent such failures, as no group has infinite funds, time, or expertise. While acknowledging such limitations, we identify two categories of "showstopper" vulnerabilities that would render *any* election authority incapable of preventing or remediating catastrophic failures.

1. **Scalable attacks:** If the cost for an adversary to tamper with the election is trivial as compared to a defender's ability to prevent such attacks from happening, attempts to prevent, remediate, or even discover the failures could be impossible in-practice. In other words, "Wholesale" attacks can be cheaper and more dangerous than "retail" attacks.
2. **Undetectable attacks:** If an attacker can alter the election outcome without any risk of the modification being caught, the attack is impossible to remediate.

The rest of this section examines how any mobile or internet voting system that lacks software software-independence suffers from *both* types of showstopper vulnerabilities, allowing attackers to remotely alter votes at larger scales with lower chance of detection than with other methods of attack.

2.1 Systems attacks

We use the term *device exploitation* to refer to attacks in which an adversary is able to modify a computer's hardware, software, or other infrastructure in order to achieve some nefarious goal.

When a system has been exploited, an attacker has complete control over what the voter sees and does. In the context of a voting machine, the attacker can prevent the user from being able to cast a vote, deceive her about any aspect of the voting process, expose her preferences to the general public, or degrade the experience to deter them from voting at all.

Exploitation is often imperceptible to a user, and can often be done so undetectably that a forensic examination of the device will not reveal malware's presence. For example, ShadowWalker, a particularly advanced example, exists only in memory, and

cannot be examined by the most privileged levels of the operating system [60]. Such malware is difficult to detect and, after the fact, may remove itself from the system without leaving a trace.

Worse, *any* communication with the outside world can lead to exploitation, so the device need not be directly connected to the Internet; malware has been known to jump such "airgaps" via USB and other removable media [21].

Systems attacks are incredibly scalable and cost-effective. It might be surprising to learn that, on the scale of elections, attack operations can be very inexpensive. As of 2012, an unpatched "zero day" vulnerability for Android cost roughly $60,000 [33]. Hypothetically, let us be extremely conservative and assume that the cost of weaponizing, testing, and leveraging the exploit increases the price by two orders of magnitude – $6,000,000. Although this may seem like a huge sum, it is important to note that (as a point of comparison), the total campaign expenditure for one candidate in the 2016 US Presidential election was roughly $768 million [47]. Compared to the research and development expenditure of a nation-state's intelligence apparatus, this cost would likely be negligible.

Once prepared, a vulnerability may be leveraged many times, and, depending on the target, a single use could affect any number of votes. Attacking centralized sources of vulnerability like the voting machine manufacturer or the voter registry, (as has happened in the 2016 election [37]), likely requires compromising few servers, and enables an attacker to quietly alter the election.

Devices themselves are vulnerable, and digital-only defenses are lacking. The systems security properties of modern computers require trusting *many* organizations to behave correctly. System flaws might be introduced by the voting software vendor, the hardware vendor, the manufacturer, or any of the third parties that maintain code for these organizations. Someone using a phone to vote is not just trusting the mobile device vendor, but the many hardware companies that wrote drivers for the device, the baseband processor, the authors of third-party code in the voting software, the manufacturer of the physical device, and the network or any other systems that the device relies upon to cast the vote. Such requirements are further impacted by geopolitical concerns: e.g., where, exactly are most devices manufactured, and who controls the voter's network?

The use of cryptography does not prevent most systems bugs from being exploitable, and, conversely, systems flaws often used to *break cryptographic guarantees*. Implementation of cryptographic code is difficult and subtle [4], and there are numerous examples of systems details breaking cryptography [3, 13].

2.2 Examples of attacks on voting systems

Researchers have repeatedly shown that polling-place electronic-only voting devices are incredibly vulnerable, even without direct connection to the Internet. For example, a 2006 paper demonstrated that the voting system used by much of Maryland and Georgia was incredibly insecure and easily exploited [22], and more recent analyses have show that such systems have not improved [11].

Internet-connected electronic voting has also been attempted and shown to be equally vulnerable. Analyses have been performed on Internet voting systems in Estonia [61], Washington DC [73], and Switzerland [42], all of which were found to be vulnerable to attackers causing catastrophic failures.

Alarmingly, there is significant evidence that election systems have been penetrated by foreign adversaries. For example, it is publicly known that the Russian government has infiltrated voter registration databases related to Florida and Illinois [37], and there are indications of similar issues in Georgia [76].

2.3 Mail-in ballots vs. electronic voting

In cases where it is impossible for a voter to access the polls, it may be necessary for the election authority to provide a solution that involves remote voting, for example, in the case of mail-in ballots for overseas military and other absentee voters.

However, the risks discussed throughout this sec-

tion militate in favor of (1) limiting remote voting to the settings where there is no feasible alternative, and (2) in such cases, using mail-in ballots and not mobile or internet voting systems. While mail-in ballots do enable vote selling and coercion much like online voting systems, they are still far less susceptible to large-scale covert attacks than online voting.

Destroying a mail-in ballot generally requires physical access, and large-scale efforts must target ballots across post offices which are geographically and operationally diverse — a very different task from exploiting a single vulnerability that could stealthily affect millions of devices with practically the same effort as one device. As a result, attacks against mail-in ballots are less likely to be scalable or to go undetected than attacks against purely electronic systems.

2.4 A note on end-to-end verifiable voting

There are some promising recent technologies called *end-to-end verifiable* (E2E-V) voting systems (e.g., [1, 2, 9, 15]), which use cryptographic techniques and post encrypted ballots on a public bulletin board[9] such that voters can verify whether their vote was included in the final tally. End-to-end verifiability can be a desirable feature to add to either paper-ballot-based or electronic-only voting systems, but does not solve the major problems described in this section. Thus, any system that is electronic only, even if end-to-end verifiable, is still unsuitable for use in consequential political elections in the foreseeable future. (Paper seems at a minimum necessary to print receipts in an E2E-V voting system.) The U.S. Vote Foundation has noted the promise of E2E-V methods for improving the security of online voting, but has issued a detailed report recommending avoiding their use for online voting unless and until the technology is far more mature and tested [26].

[9] Section 3 discusses how blockchains could be used to implement a public bulletin board. However, we will see blockchain technology does not add anything *beyond* a way of implementing a public bulletin board, and as such, does not help solve existing issues with E2E-V voting systems.

2.5 The importance of transparency

As we have seen in this section, software is complicated and it is very hard to get it right. Commercial software typically contains 1–25 bugs for every thousand lines of code. Moreover, if the software implements security mechanisms, it should not only be correct but provide credible assurance of secure operation to those who depend on those mechanisms. Not only is the design challenging to get right, but the implementation can be particularly challenging to get right if the adversary may corrupt insiders (such as software developers) in the supply chain.

Today, it is best practice, including among cryptocurrency implementations, to adopt *open-source* development methods.[10] Disclosed-source implementations allow one to gain substantial (though not necessarily complete) confidence that the implementation contains no serious bugs or security holes. In fact, disclosing security-critical system designs for inspection by experts and even "the enemy" has been considered a good security practice since as early as the 19th century [40]. This may be surprising given the intuition that a secret system design is harder for an adversary to figure out; but lack of scrutiny also makes it much easier for security vulnerabilities to go on unnoticed and unaddressed.

Thus, security-critical software that is closed-source carries much higher risk and uncertainty than disclosed-source alternatives. Accordingly, voting systems should favor disclosing system designs and code whenever possible.

That said, transparency is not a panacea. It is generally the case that one cannot verify that the code running on a given machine is actually the compiled version of the open-source software that was reviewed; devising such verification methods is an area of active ongoing research.[11] So, while

[10] By "open-source" here we really mean "disclosed-source," where the source code is open for all to read but changes may be controlled. See Wallach [70] for a more detailed discussion of open source vs. disclosed source in voting systems.

[11] For example, Fink et al. [24] study the potential use of *trusted platform modules* (TPMs) to mitigate concerns that the software running is not the software that is supposed to be running. Of course, one still has the concern that the TPM system itself is free from bugs, and in any case this doesn't address the correctness of the voting system software.

transparency (disclosed software and good cryptographic protocol documentation) seems necessary for security, it is by no means sufficient.

3 Blockchains as a ballot box

A number of recent proposals purport to use blockchain technology to add security to electronic voting [25, 68, 69]. We show that blockchains do not address the issues in the previous section and might introduce new problems.

This section begins by reviewing blockchain technology (§3.1,§3.2). Those familiar with blockchain technology may wish to skim or skip these subsections. Then §3.3 re-emphasizes and gives examples illustrating that blockchain voting is still online voting, and thus suffers the same vulnerability to catastrophic failures that was described in §2. §3.4 discusses how blockchain-based electronic voting could create yet further problems for election systems, beyond what was discussed in §2. Finally, §3.5 describes voting used *within* blockchain technology, which we distinguish from voting in political elections.

3.1 Blockchain technology overview

The term *blockchain*, confusingly, has been used to refer to a wide range of technologies, including distributed databases, hashing, digital signatures, and sometimes even multiparty computation and zero-knowledge proofs. All of these technologies individually pre-date the first blockchain, Bitcoin .

A blockchain implements what in cryptography has been known as a *public bulletin board*. It is a linear ordering of data with the following properties: it is append-only; data can only be added to the board, not removed. It is public and available: everyone can read the data on the board, and every reader sees a common prefix of the same ordering.

For example, Bitcoin's blockchain is a list of transactions. Users can add transactions to the blockchain, and read the list of transactions to find out who owns which bitcoins.

Blockchains implement validation rules: by consensus, only data with a certain format is appended to the blockchain. For example, in all cryptocurrencies, transactions transferring money need to pass a set of validity checks or they will not be appended: The sender must have the funds to send, and the transaction must show correct authorization to move the funds.

All of this functionality is provided under certain assumptions; in the case of Bitcoin it is assuming the majority of the mining hash power is honest. In other blockchains, it might be that at least two thirds of the participants are honest. If these assumptions do not hold true, the blockchain might not be able to provide its availability, linear ordering, and common prefix guarantees.

3.2 How to achieve a blockchain interface

In order to achieve the public bulletin board functionality, blockchains typically operate in the following manner. A network of computers run a common piece of software to agree on an ordered log of data. Users submit new data with digital signatures, and the software running on the network of computers enforces validation rules, like users cannot create new coins outside the specified monetary policy. It also runs a protocol to agree on the continuing log of data, and chains the data together using hashes so that one cannot tamper with data in the past without being detected.

Consensus. Distributed consensus is the problem of many computers trying to agree on a value in the presence of failures. Before Bitcoin, the designers of consensus protocols assumed that the set of participants was known, and relied on sending messages to everyone. The core innovation behind Bitcoin is a *permissionless* distributed consensus protocol that is secured using incentives, known as Nakamoto consensus [45]. Bitcoin uses a technique called *proof-of-work* [6, 18] to select the next block in the blockchain; in Bitcoin the "work" is producing a preimage of a partially-fixed hash. Participants who do this work are known as *miners*. The first miner to find a preimage broadcasts their block to the Bitcoin network and, once the block is accepted, is paid in Bitcoin specified in the block they produced; this is called the block reward. The block

reward consists of both newly minted Bitcoin and the transaction fees of the transactions included in the block.

Miners must expend a lot of computational cycles to find this preimage; this makes proof-of-work energy intensive and its cost dominated by operational costs. Because of this, most miners have gravitated to geographical locations with cheap energy, and many large miners are based in China. The security of Nakamoto consensus relies on the assumption that the majority of the mining power is honest.

Newer cryptocurrencies have implemented a newer type of consensus protocol called *proof-of-stake*, which is much less energy intensive. These protocols are more like traditional consensus protocols except the set of participants is determined by who holds stake, or coins, in the system.

The advent of permissionless protocols has caused many people to take a second look at distributed databases where nodes in the database are run by different organizations. These types of databases are sometimes called *permissioned* blockchains because similar to permissionless blockchains they are a verifiable log of records, but the set of participants is limited and determined ahead of time (a node requires permission to join the blockchain system). These protocols help with fault tolerance, and can even tolerate some fraction of the nodes in the system (typically up to a third) being malicious. This distributed database technology can help make a centralized database more resilient to failures; however, we shall see that this does not address the core problems with internet and mobile voting.

Authentication. In all of these systems, users create a digital signature to indicate consent to authorize a transaction to add to the blockchain, perhaps spending coins. Other nodes in the network validate signatures and check that each batch of transactions maintains financial invariants, like the spender has the funds to spend, or that coins are created upon an agreed upon issuance schedule. In a blockchain without a coin, nodes might validate other application-specific rules.

Smart contracts. Blockchains support more complex operations than just transferring coins. Coins can be transferred conditionally, according to scripts or smart contracts. For example, in Bitcoin, coins can be locked up for a period of time or require multiple signatures to spend. Blockchains like Ethereum support even richer smart contracts—the Ethereum network functions like a single, global computer running different smart contract programs.

Transaction secrecy. Blockchains, by default, do not keep transaction details secret. A key component of blockchain technology is that transactions are verifiable, and public verifiability seems at odds with secrecy. In the case of permissioned blockchains, the participants running the blockchain can restrict read access to the blockchain. This can be helpful to limit data leakage, but it comes with a sacrifice: now those who do not have read access cannot download and verify the blockchain. In a permissionless blockchain, there is no restricted set of participants so the entire transaction history is public to everyone. Some cryptocurrencies use *zero-knowledge proofs* to hide information about transactions (the participants in the transaction and the amount) while still maintaining public verifiability. A zero-knowledge proof shows that some statement is true without revealing why that statement might be true. For example, using a zero-knowledge proof, I can convince you that I know the answer to a specific Sudoku puzzle without revealing the actual answer. Zero-knowledge proofs were invented many decades before blockchain technology and might be useful in constructing electronic voting systems, though they are not enough alone.

Applications. Blockchains can be used for more than cryptocurrency. For example, IBM is using the Hyperledger Fabric blockchain to record the provenance of food as it travels through a food supply chain. Participants include producers, suppliers, manufacturers, and retailers and the goal is to "provide authorized users with immediate access to actionable food supply chain data, from farm to store and ultimately the consumer." Everledger is a company aiming to track diamonds using blockchain

technology . Note that these applications require entities to make in-blockchain claims about assets and operations in the real world.

3.3 Blockchain technology applied to voting

Bitcoin, the first example of blockchain technology, operates in a highly adversarial environment—anyone can download the software and join the network, including attackers. The idea behind Bitcoin is that participants sign transactions to indicate consent to transfer, and are constantly downloading and validating the blockchain to ensure that rules are being followed and their coins are valid. Blockchains use consensus protocols to avoid a single point of failure; these protocols can tolerate a small number of participants acting maliciously.

Some of these ideas seem as though they might be helpful for electronic voting—for example, using public key cryptography to make it difficult to forge votes, and using hashing and a distributed consensus protocol to maintain a ledger of votes so that an attacker cannot tamper with the history of votes without co-opting much of the network. However, it is extremely challenging to make these techniques work reliably in practice: blockchain voting is still electronic voting, and blockchains do not address the problems described in the previous section. In particular, any blockchain voting system is still vulnerable to catastrophic failures, and *the cryptographic and consensus guarantees of blockchains do not prevent catastrophic failures.*

The following is an example of an approach one might consider when designing a blockchain-based voting system, and how it fails to address several issues. This design does not consider every detail of implementing a voting system on a blockchain and is not exhaustive. Instead, it demonstrates issues that would apply to many designs.

Coins as votes. Here is a strawman using a blockchain as a ballot box: Candidates register with the voting authority with their public keys. The voting authority, which maintains a voter registry, gets each registered user to create a public/private key pair, and each user sends their public key to the registry. Then, the voter registry spends one coin each to each public key. In order to vote, each user spends their coin to the candidate of their choice. After a time period, everyone can look at the blockchain, total up how much coins each candidate has, and select the one with the most coins as the winner.

This strawman design has several problems. First of all, it does not provide a secret ballot: all votes are public, and users can prove to a third party how they voted, so they could conceivably be coerced to vote a certain way. Users could sell their votes.

Second, this design relies on users being able to get their votes on the blockchain in the given election time period. The vote tallier cannot wait for *all* users to spend their coins because that means a single user could prevent the election from finishing; there must be some cutoff point. Public blockchains, in particular, are limited in throughput and require fees to submit transactions. During times of high transaction rates, fees can get quite high, and transactions can be delayed. An attacker willing to spend enough money could flood the blockchain with transactions to drive up fees and keep users from voting until after the cutoff point has passed.

Third, the design only works if the blockchain properly implements the public bulletin board interface. If the blockchain is compromised (for example, a majority of the miners or validators collude) then they could create multiple versions of the blockchain to show different people, sowing discord. Or, they could censor certain users' votes. Several cryptocurrencies have suffered from these types of attacks, where their blockchains have been rewritten . Blockchains are often referred to as "immutable," but these attacks show that this is not always true in practice.

Fourth, security of this strawman hinges on private keys. If a user loses her private key, she can no longer vote, and if an attacker obtains a user's private key they can now undetectably vote as that user. Many users have lost access to their private keys and thus have lost their cryptocurrency. This has even happend to cryptocurrency exchanges, which have lost hundreds of millions of dollars worth of cryptocurrency to attackers or through bad key management . Blockchains cannot help if

a user's keys are compromised; in fact blockchain-based systems seem to *require* using public key cryptography. A blockchain-based electronic voting system would also need to maintain and run a secure public key directory.

Finally, all of the above depends on secure software and hardware on the part of the user. If a user's voting device (probably a mobile phone) is compromised, so is their vote.

A note on permissioned blockchains. One might think of using a permissioned blockchain, instead, at least to solve the first and second issues. However, a permissioned blockchain system would still suffer from the remaining issues, and, depending on how it is implemented, new ones—if users cannot read the permissioned blockchain and verify that their votes were counted, it does not implement a verifiable tally. (If everyone could read the blockchain, then they could prove how they voted by pointing it out and it would not be a secret ballot). There are even fewer, more homogenous servers to compromise compared to large public blockchain instances. It is possible they could all be compromised, especially if they run on the same operating system or run the same software, which most proposed blockchain-based voting systems do. Permissioned blockchains also do nothing to address the issues around security of private keys or the security of the software and hardware on the user devices.

Zero-knowledge proofs for secret ballots. There are cryptocurrency schemes which keep the content of transactions secret while still allowing everyone to verify that certain financial invariants are maintained, getting around the challenge described above between secrecy and public verifiability. These schemes make use the zero-knowledge proof primitive described in §3.2. For example, Zerocash [55] and its subsequent implementation in the cryptocurrency Zcash [35] provide *shielded* transactions, which are secret and do not reveal amounts, senders, or receivers. Despite this, everyone can still verify these transactions' financial invariants much like with public blockchain transactions, even though they are secret.

One could apply such a scheme to the strawman to make use of shielded transactions. Implemented correctly, this could provide a secret ballot, and some E2E-V schemes in fact use zero-knowledge proofs. However, there are two new issues that such a design introduces: First of all, we have added in fairly complex, new cryptography to our system. This is less understood and vetted than older, more conservative cryptographic primitives and new bugs are still being discovered. In 2018 a critical bug in Zcash was discovered that would have allowed an attacker to undetectably counterfeit Zcash coins [64].

Second, because this cryptography is complicated and exotic there is little chance of the general public understanding how it works; to an average citizen, it might seem like black magic, or something they must take on faith. It is important for democracy for the public to believe in the correctness of election technology so that they believe in the results of the election.

Importantly, elections are much higher-stakes than cryptocurrency. An attack on many cryptocurrency users would cause monetary loss, an attack on many voters can cause government regime change.

3.4 New problems blockchains introduce

In addition to all the issues with mobile and internet voting that still apply to blockchain technology, a blockchain-based voting system would introduce new security concerns. Blockchains are designed to be decentralized, or run by multiple actors. Decentralized technology can introduce inefficiencies when compared to centralized solutions. Also, by their nature, protocol governance requires coordination and can be difficult. We saw examples where this caused a lot of strife . Importantly, blockchain technology introduces a lot of *complexity* in the software and management of the software. Distributed consensus protocols and cryptographic systems are very difficult to implement correctly. Any additional complexity in voting means more ways for things to potentially go wrong.

This additional complexity also introduces problems with fixing bugs and deploying new software. It takes more time to deploy security fixes in a decentralized system than in a centralized one, mean-

ing blockchain systems can be vulnerable for longer periods of time than their centralized counterparts. In a critical application like voting, it is important to be able to move quickly to fix bugs.

Other work has proposed frameworks for determining when an application is a good fit for blockchain technology [53, 56, 74]. Though voting requires auditing, it does not warrant the complexity introduced by a technology supporting shared governance and shared operation. Voting is inherently centralized (with a central organization, the government, that is in charge of election procedures and eligibility to vote).

Despite the first blockchain, Bitcoin, launching in 2009, it took several years to gain users and for developers to gain some experience securing the platform. The technology is very much still new and under development. It usually doesn't make sense to use a new cryptographic protocol for critical infrastructure until it has been well-tested in industry for many years. Blockchain technology has not yet reached this point.

3.5 Voting within blockchains

Blockchain protocols and smart contracts sometimes employ voting *within* the blockchain or contract application. For example, in EOS, token holders can vote for validators to participate in the consensus network protocol and select blocks. It is important to note the use of the term "voting" here; this is not a political election, it is a consensus protocol. A maliciously elected EOS validator could slow down validation or validate incorrect blocks, potentially affecting holders of the EOS cryptocurrency. Malicious validators in political elections could do much worse.

Some smart contracts let token holders vote on contract outcomes. For example, Augur is a protocol for creating prediction markets which run on Ethereum where users can bet on the outcomes of sporting events, market movements, weather, and more [50]. Augur has a built-in token called REP. REP token holders stake their tokens to vote on real-world outcomes and report them into the smart contract. REP holders are responsible for participating in contract disputes and will be penalized (they will automatically lose some of their REP) if they do not participate. Note that this process does not fulfill any of the requirements for secure voting.

3.6 Summary

In summary, a bulletin-board like interface combined with encryption for secrecy seems helpful as a voting ballot box, but these techniques still do not address several fundamental security issues with electronic voting. It remains unclear what type of role decentralization should play; on the one hand, systems with a small number of homogenous nodes might be more likely to suffer from compromise. On the other hand, voting is inherently a centralized process, and decentralized systems come with many drawbacks, including potential congestion and difficulty in upgrading.

4 Critical Questions

As a short article like this can not provide a comprehensive guide to all of the issues that might be raised about "voting on the blockchain," the next subsection provides the reader with some questions that should be asked about any such proposal.

We list here some high-level questions relating to the *security* of any proposed voting system. These questions do not focus on other important aspects of voting systems (e.g., usability, cost, accessibility, etc.). While good security cannot be achieved simply by "passing a checklist," a good set of questions can illuminate gaps in reasoning, poor assumptions, implementation problems, etc.

Stakeholders and Adversaries Who are the voting system *stakeholders*? Who are the potential *adversaries*? These are often the same! They include:

- Candidates
- Voters
- Election officials
- Auditors
- The public (including observers who might not be voters)
- Foreign observers
- System designers and vendors (who supply software or hardware components, or who pro-

vide operational assistance in the running of an election)

Security objectives.

- What security properties is the system intended to have? What should an Adversary not be able to do (for many variations on the notion of Adversary, including the participation of corrupt "insiders")?
- What is the threat model? For high-stakes political elections the threat model should include at least:
  - Compromise of a device's hardware and/or software, possibly via supply-chain attacks
  - Failure to properly record a voter's choices
  - Tabulation errors
  - Selling of votes
  - Corruption of evidence trail
  - Ballot "stuffing" (extra ballots) or ballot destruction
- What kinds of plausible attacks are not considered? (Does the security of the the system depend on "trusted hardware" or "trusted software"?)
- How many people would an Adversary have to corrupt in order to steal an election?

Security mechanism design

- What security mechanisms proposed in the system design?
- Are those mechanisms designed to *prevent* security violations, or to just *detect* such violations?
- What happens when a security violation is detected?
- Do the proposed mechanisms rely on particular behaviors by certain parties (voters, election officials, etc.) to be effective?
- If voting system computers or devices are compromised, what is the worst-case effect it could have on the reported election outcome?
  - Would that effect be reliably detectable? How?
- What mechanisms enable voters and observers to verify that the system works as it is intended to, and that the outcomes produced have not been affected by Adversarial behavior?

Evidence-based elections

- What evidence does the system produce supporting the reported outcome?
- Why should that evidence be considered trustworthy? Are any assumptions about the correct operation of the system required? Does concluding that the evidence is trustworthy require trusting that one or more computer systems are operating correctly? If so, are those assumptions credible and/or verifiable?
- Is that evidence auditable? What forms of audits are supported and what assurance do they provide, to whom?

Verification

- Who can verify the system's design and operation? Neutral third parties? The Federal certification process? (Based on the VVSG?)
- How many different parties can verify? What is their expertise/interests?
- What credible assurance comes out of these verification processes, to whom, about what?
- Is the assurance about a sample implementation before the election, or about the operation of the system during the election? (More succinctly, does one verify the system, or does one verify the outcome?)
- What oversight/verification is there that the outsourced components (people and software) work properly?
- What if a bug is found in the code? How do you discover it? How do you address it?

Cryptography If cryptography is used:

- How are keys managed?
- What happens if one or more keys are compromised?
- Can parties "reset their keys" (choose new keys to replace ones that have been lost or compromised)?

Remote voting If voting is done remotely:

- What credentials does the voter have that allows him/her to vote? How do they obtain those credentials?

Operation

- What instructions are given to voters/election officials/others to manage ex-

ceptional/erroneous situations? E.g., what is a voter supposed to do if they see an incorrect printout or a candidate missing from a ballot? What evidence enables the error to be confirmed?
- How much outsourcing to vendors is involved in the operational aspects of the election? Can the election outcome be trusted if the vendors are not trusted?
- What if the system is discovered to be malfunctioning during the election? How do you discover it? How do you address it?
- It's easy to design a system that works fine if everything goes as expected. How does the proposed system handle unexpected faults and security violations?
- Could a voter credibly prove how they voted to a third party?

5 Related work

The United States National Academies has recently produced an excellent report [46] providing an overview of election security. We note that this report includes a section on "Internet Voting" that briefly discusses whether blockchains can be helpful in providing additional security, which concludes (page 104) that

> "While the notion of using a blockchain as an immutable ballot box may seem promising, blockchain technology does little to solve the fundamental security issues of elections, and indeed, blockchains introduce additional security vulnerabilities."

Other researchers in the computer security and blockchain fields have written about the risks of blockchain voting in publications such as Slate [31] and The Conversation [39]. A collection of useful links to online resources related to blockchains and voting is available on Duncan Buell's website [14]. Finally, we can't resist mentioning the lovely XKCD comic [44] on the topic of blockchain voting!

6 Conclusion

A summary of this article's takeaways follows.

1. **Blockchain technology does not solve the fundamental security problems suffered by all electronic voting systems §3.** Moreover, blockchains may introduce new problems that non-blockchain-based voting systems would not suffer from.

2. **Electronic, online, and blockchain-based voting systems are vastly more vulnerable to catastrophic failures than available paper-ballot-based alternatives (§2).** Moreover, given the state of the art in computer security, they will continue to be so for the foreseeable future.

3. **Adding new technologies to systems may create new potential for attacks.** Particular caution is appropriate in security-critical applications, especially where political pressures may favor an expedited approach. (§3.4).

The article has also provided a **collection of critical questions** intended as a reference point for evaluating any new voting system proposal from a security perspective (§4), and provided references for further reading on this topic (§5).

References

[1] Ben Adida. "Advances in Cryptographic Voting Systems". PhD thesis. MIT, 2006.

[2] Ben Adida. "Helios: Web-based Open-Audit Voting". In: *Proceedings of the 17th USENIX Security Symposium, July 28-August 1, 2008, San Jose, CA, USA*. Ed. by Paul C. van Oorschot. USENIX Association, 2008, pp. 335–348. ISBN: 978-1-931971-60-7. URL: http://www.usenix.org/events/sec08/tech/full_papers/adida/adida.pdf.

[3] David Adrian, Karthikeyan Bhargavan, Zakir Durumeric, Pierrick Gaudry, Matthew Green, J. Alex Halderman, Nadia Heninger, Drew Springall, Emmanuel Thomé, and Luke Valenta. "Imperfect forward secrecy: How Diffie-Hellman fails in practice". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 5–17.

[4] Ross J. Anderson. "Why Cryptosystems Fail". In: *Commun. ACM* 37.11 (1994), pp. 32–40. DOI: 10.1145/188280.188291. URL: https://doi.org/10.1145/188280.188291.

[5] Andrew W. Appel, Richard A. DeMillo, and Philip B. Stark. *Ballot-Marking Devices (BMDs) Cannot Assure the Will of the Voters*. 2019.

[6] Adam Back et al. "Hashcash-a denial of service counter-measure". In: (2002).

[7] Gregory Barber. *Wouldn't It Be Great If People Could Vote on the Blockchain?* https://www.wired.com/story/wouldnt-it-be-great-if-people-could-vote-on-blockchain. 2019.

[8] Matthew Beedham. *Japan is experimenting with a blockchain-powered voting system*. The Next Web. https://thenextweb.com/hardfork/2018/09/03/japan-city-blockchain-voting. 2018.

[9] Josh Benaloh, Ronald L. Rivest, Peter Y. A. Ryan, Philip B. Stark, Vanessa Teague, and Poorvi L. Vora. *End-to-end verifiability*. Apr. 15, 2015.

[10] Matt Blaze, Jake Braun, Harri Hursti, David Jefferson, Margaret MacAlpine, and Jeff Moss. *DEF CON 26 Voting Village: Report on Cyber Vulnerabilities in U.S. Election Equipment, Databases, and Infrastructure*. https://www.defcon.org/images/defcon-26/DEF%20CON%2026%20voting%20village%20report.pdf. 2018.

[11] Matt Blaze, Harri Hursti, Margaret MacAlpine, Mary Hanley, Jeff Moss, Rachel Wehr, Kendall Spencer, and Christopher Ferris. *DEF CON 27 Voting Machine Hacking Village*. https://media.defcon.org/DEF%20CON%2027/voting-village-report-defcon27.pdf. 2019.

[12] Matt Blaze, Jake Braun, Harri Hursti, Joseph Lorenzo Hall, Margaret MacAlpine, and Jeff Moss. *DEFCON 25 Voting Machine Hacking Village: Report on Cyber Vulnerabilities in U.S. Election Equipment, Databases, and Infrastructure*. https://www.defcon.org/images/defcon-25/DEF%20CON%2025%20voting%20village%20report.pdf. 2017.

[13] David Brumley and Dan Boneh. "Remote timing attacks are practical". In: *Computer Networks* 48.5 (2005), pp. 701–716.

[14] Duncan Buell. *Blockchains and Voting*. https://cse.sc.edu/~buell/blockchain-papers.

[15] David Chaum. "Secret-Ballot Receipts: True Voter-Verifiable Elections". In: *IEEE Security & Privacy* 2.1 (2004), pp. 38–47. DOI: 10.1109/MSECP.2004.1264852. URL: https://doi.org/10.1109/MSECP.2004.1264852.

[16] Kevin Collier. *FBI investigating alleged hacking attempt into mobile voting app during 2018 midterms*. CNN. https://www.cnn.com/2019/10/01/politics/fbi-hacking-attempt-alleged-mobile-voting-app-voatz/index.html [https://perma.cc/DF56-F9B6]. Oct. 2019.

[17] Régis Dandoy. "The Impact of e-Voting on Turnout: Insights from the Belgian Case". In: Apr. 2014, pp. 29–37. ISBN: 978-3-907589-17-5. DOI: 10.1109/ICEDEG.2014.6819940.

[18] Cynthia Dwork and Moni Naor. "Pricing via processing or combatting junk mail". In: *Annual International Cryptology Conference*. Springer. 1992, pp. 139–147.

[19] Equifax. *Equifax Releases Details on Cybersecurity Incident, Announces Personnel Changes*. https://investor.equifax.com/news-and-events/news/2017/09-15-2017-224018832 [https://perma.cc/6AD3-P7LV]. Sept. 2017.

[20] *Equifax Data Breach Settlement*. FTC. https://www.ftc.gov/enforcement/cases-proceedings/refunds/equifax-data-breach-settlement [https://perma.cc/38BK-RS33].

[21] Nicolas Falliere, Liam O Murchu, and Eric Chien. "W32. stuxnet dossier". In: *White paper, Symantec Corp., Security Response* 5.6 (2011), p. 29.

[22] Ariel J. Feldman, J. Alex Halderman, and Edward W. Felten. "Security Analysis of the Diebold AccuVote-TS Voting Machine". In: *2007 USENIX/ACCURATE Electronic Voting Technology Workshop, EVT'07, Boston,*

MA, USA, August 6, 2007. Ed. by Ray Martinez and David A. Wagner. USENIX Association, 2007. URL: https://www.usenix.org/conference/evt-07/security-analysis-diebold-accuvote-ts-voting-machine.

[23] Jacqueline Fernandez. *County To Survey Voters On Proposed Changes*. Los Angeles Wave Newspapers. http://wavenewspapers.com/county-to-survey-voters-on-proposed-changes/. 2018.

[24] Russell A. Fink, Alan T. Sherman, and Richard Carback. "TPM Meets DRE: Reducing the Trust Base for Electronic Voting Using Trusted Platform Modules". In: *Trans. Info. For. Sec.* 4.4 (Dec. 2009), pp. 628–637. ISSN: 1556-6013. DOI: 10.1109/TIFS.2009.2034900. URL: https://doi.org/10.1109/TIFS.2009.2034900.

[25] *Follow My Vote*. https://followmyvote.com.

[26] Overseas Vote Foundation. *The Future of Voting: End-to-End Verifiable Internet Voting — Specification and Feasibility Study*. (One of the authors, Rivest, was on the Advisory Council for this report.) July 2015.

[27] Pierrick Gaudry. "Breaking the encryption scheme of the Moscow internet voting system". In: *CoRR* abs/1908.05127 (2019). arXiv: 1908.05127. URL: http://arxiv.org/abs/1908.05127.

[28] Micha Germann and Uwe Serdült. "Internet voting and turnout: Evidence from Switzerland". In: *Electoral Studies* 47 (Mar. 2017). DOI: 10.1016/j.electstud.2017.03.001.

[29] Alexander Golovnev. "An Attack on the the Encryption Scheme of the Moscow Internet Voting System". In: *CoRR* abs/1908.09170 (2019). arXiv: 1908.09170. URL: http://arxiv.org/abs/1908.09170.

[30] Nicole Goodman and Leah C. Stokes. "Reducing the Cost of Voting: An Evaluation of Internet Voting's Effect on Turnout". In: *British Journal of Political Science* (2018), 1–13. DOI: 10.1017/S0007123417000849.

[31] Rachel Goodman and J. Alex Halderman. *Internet Voting is Happening Now*. https://slate.com/technology/2020/01/internet-voting-could-destroy-our-elections.html. Jan. 2020.

[32] Yael Grauer. *What Really Happened With West Virginia's Blockchain Voting Experiment?* https://slate.com/technology/2019/07/west-virginia-blockchain-voting-voatz.html [https://perma.cc/H9M5-YJSV]. July 2019.

[33] Andy Greenberg. *Shopping For Zero-Days: A Price List For Hackers' Secret Software Exploits*. en. URL: https://www.forbes.com/sites/andygreenberg/2012/03/23/shopping-for-zero-days-an-price-list-for-hackers-secret-software-exploits/ (visited on 05/23/2019).

[34] U.S. Department of Homeland Security. *Election Security*. https://www.dhs.gov/topic/election-security [https://perma.cc/2PRL-EMYS].

[35] Daira Hopwood, Sean Bowe, Taylor Hornby, and Nathan Wilcox. "Zcash protocol specification". In: *Technical report 2016–1.10. Zerocoin Electric Coin Company* (2016).

[36] *H.R. 2722 — SAFE Act (Securing America's Federal Elections Act)*. Congress.gov. Introduced by Rep. Zoe Lofgren on May 5, 2019. Passed the House on June 27, 2019. Received in the Senate on June 28, 2019. https://www.congress.gov/bill/116th-congress/house-bill/2722 [https://perma.cc/NA6K-FMVX].

[37] Robert S. Mueller III. *Report on the Investigation into Russian Interference in the 2016 Presidential Election ("The Mueller Report")*. U.S. Department of Justice. Mar. 2019.

[38] David Jefferson, Duncan Buell, Kevin Skoglund, Joe Kiniry, and Joshua Greenbaum. *What We Don't Know About the Voatz "Blockchain" Internet Voting System*. https://cse.sc.edu/~buell/blockchain-papers/documents/WhatWeDontKnowAbouttheVoatz_Blockchain_.pdf. 2019.

[39] Ari Juels, Ittay Eyal, and Oded Naor. *Blockchains won't fix internet voting security – and could make it worse*. The Conversation. http://theconversation.com/blockchains-wont-fix-internet-voting-security-and-could-make-it-worse-104830 [https://perma.cc/2VQQ-25H9]. Oct. 2018.

[40] Auguste Kerckhoffs. "La Cryptographie Militaire". In: *Journal des sciences militaires* IX (1883), pp. 5–83.

[41] Julia Krivonosova. *Internet voting in Russia: how?* Medium. https://medium.com/@juliakrivonosova/internet-voting-in-russia-how-9382db4da71f [https://perma.cc/EP9B-K6B7]. July 2019.

[42] Sarah Jamie Lewis, Olivier Pereira, and Vanessa Teague. "How not to prove your election outcome". en. In: (Mar. 2019), p. 11.

[43] Glen Mills. *Utah County Clerk says mobile voting pilot program was a success*. ABC4. https://www.abc4.com/news/utah-county-clerk-says-mobile-voting-pilot-program-was-a-success. 2019.

[44] Randall Munroe. *Voting Software*. https://xkcd.com/2030. Aug. 8, 2018.

[45] Satoshi Nakamoto et al. "Bitcoin: A peer-to-peer electronic cash system". In: (2008).

[46] National Academies of Sciences, Engineering, and Medicine. *Securing the Vote: Protecting American Democracy*. Washington, DC: The National Academies Press, Sept. 6, 2018.

[47] Niv M. Sultan. *Election 2016: Trump's free media helped keep cost down*. en-US. Apr. 2017. URL: https://www.opensecrets.org/news/2017/04/election-2016-trump-fewer-donors-provided-more-of-the-cash/ (visited on 05/23/2019).

[48] Official Website of the Mayor of Moscow. *Взломать нельзя, тестировать: программисты проверят надежность электронного голосования*. https://perma.cc/5JCY-S5EA [https://perma.cc/5JCY-S5EA].

[49] Official Website of the Mayor of Moscow. *Электронные выборы в Московскую городскую Думу*. https://www.mos.ru/city/projects/blockchain-vybory [https://perma.cc/XZB4-FD9F].

[50] Jack Peterson and Joseph Krug. "Augur: a decentralized, open source platform for prediction markets". In: *arXiv preprint arXiv: 1501.01042* (2015).

[51] Ronald L. Rivest. "On the notion of 'software independence' in voting systems". In: *Philosophical Transactions of the Royal Society* 366 (2008). https://royalsocietypublishing.org/doi/pdf/10.1098/rsta.2008.0149, pp. 3759–67.

[52] Ronald L. Rivest and Madars Virza. "Software Independence Revisited". In: *Real-World Electronic Voting: Design, Analysis and Deployment*. Ed. by Feng Hao and Peter Y. A. Ryan. Taylor & Francis. Chap. 1.

[53] Scott Ruoti, Ben Kaiser, Arkady Yerukhimovich, Jeremy Clark, and Robert Cunningham. "Blockchain technology: what is it good for?" In: *Communications of the ACM* 63.1 (2019), pp. 46–53.

[54] *S. 1540 — Election Security Act of 2019*. Congress.gov. Introduced by Sen. Amy Klobuchar on May 16, 2019.

[55] Eli Ben Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. "Zerocash: Decentralized anonymous payments from bitcoin". In: *2014 IEEE Symposium on Security and Privacy*. IEEE. 2014, pp. 459–474.

[56] Brian A Scriber. "A Framework for Determining Blockchain Applicability". In: *IEEE Software* 35.4 (2018), pp. 70–77.

[57] Andrew Selsky. *2 Oregon counties offer vote-by-mobile to overseas voters*. AP News. https://apnews.com/8ce0fbc400514f55839fa84fb364d7f4. 2019.

[58] Uwe Serdült, Micha Germann, Maja Harris, Fernando Mendez, and Alicia Portenier. "Who Are the Internet Voters?" English. In: *Electronic Government and Electronic Participation*. Ed. by Efthimios Tambouris and et

al. Innovation and the Public Sector. Netherlands: IOS Press, 2015, pp. 27–41. ISBN: 9781614995692. DOI: 10.3233/978-1-61499-570-8-27.

[59] Hamza Shaban. *Binance says hackers stole $40 million worth of bitcoin in one transaction*. Washington Post. https://www.washingtonpost.com/technology/2019/05/08/binance-says-hackers-stole-million-worth-bitcoin-one-transaction/. 2019.

[60] Sherri Sparks and Jamie Butler. "Shadow Walker: Raising The Bar For Windows Rootkit Detection". In: *Phrack Magazine* 0x0b.0x3d (). URL: http://phrack.org/issues/63/8.html (visited on 05/23/2019).

[61] Drew Springall, Travis Finkenauer, Zakir Durumeric, Jason Kitcat, Harri Hursti, Margaret MacAlpine, and J. Alex Halderman. "Security analysis of the Estonian internet voting system". In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014, pp. 703–715.

[62] Philip B. Stark. "There is no Reliable Way to Detect Hacked Ballot-Marking Devices". In: *ArXiv* abs/1908.08144 (2019).

[63] Katherine Stewart and Jirka Taylor. *Online Voting: The Solution to Declining Political Engagement?* https://www.rand.org/blog/2018/03/online-voting-the-solution-to-declining-political-engagement.html. 2018.

[64] Josh Swihart, Benjamin Winston, and Sean Bowe. *Zcash Counterfeiting Vulnerability Successfully Remediated*. Feb. 5, 2019. URL: \href{https://electriccoin.co/blog/zcash-counterfeiting-vulnerability-successfully-remediated/}.

[65] swissinfo.ch. *Switzerland's first municipal blockchain vote hailed a success*. https://www.swissinfo.ch/eng/crypto-valley-_-switzerland-s-first-municipal-blockchain-vote-hailed-a-success/44230928. 2018.

[66] Matt Tatham. *Identity theft statistics*. Experian. https://www.experian.com/blogs/ask-experian/identity-theft-statistics [https://perma.cc/3UEB-JLW5]. Mar. 2018.

[67] moscow technologies. *moscow-technologies / blockchain-voting*. GitHub. https://github.com/moscow-technologies/blockchain-voting [https://perma.cc/LL8M-6GN2].

[68] *Voatz*. https://voatz.com.

[69] *Votem*. https://www.votem.com.

[70] Dan Wallach. *On open source vs. disclosed source voting systems*. https://freedom-to-tinker.com/2009/04/16/open-source-vs-disclosed-source-voting-systems/. 2009.

[71] West Virginia Secretary of State's Office. *24 Counties to Offer Mobile Voting Option for Military Personnel Overseas*. https://sos.wv.gov/news/Pages/09-20-2018-A.aspx [https://perma.cc/CX3E-YBPQ]. Sept. 2018.

[72] West Virginia Secretary of State's Office. *Warner Pleased with Participation in Test Pilot for Mobile Voting*. https://sos.wv.gov/news/Pages/11-16-2018-A.aspx [https://perma.cc/7VDD-PZFP]. Nov. 2018.

[73] Scott Wolchok, Eric Wustrow, Dawn Isabel, and J. Alex Halderman. "Attacking the Washington, DC Internet voting system". In: *International Conference on Financial Cryptography and Data Security*. Springer, 2012, pp. 114–128.

[74] Karl Wüst and Arthur Gervais. "Do you need a Blockchain?" In: *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*. IEEE. 2018, pp. 45–54.

[75] *Wyden and Bicameral Coalition Introduce Bill to Require States to Secure Elections*. Ron Wyden's Official Website. https://www.wyden.senate.gov/news/press-releases/wyden-and-bicameral-coalition-introduce-bill-to-require-states-to-secure-elections-. 2019.

[76] Kim Zetter. *Was Georgia's Election System Hacked in 2016?* en. URL: https://politi.co/2moAWUS (visited on 05/23/2019).