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§

February 20, 2020 (DRAFT)

Abstract

Voters are understandably concerned about election security. News reports of possible election interference by foreign powers, of unauthorized voting, of voter disenfranchisement, and of technological failures call into question the integrity of elections worldwide.

This article examines the suggestions that "voting over the Internet" or "voting on the blockchain" would increase election security, and finds such claims to be wanting and misleading. While current election systems are far from perfect, Internet- and blockchain-based voting would greatly increase the risk of undetectable, nation-scale election failures.

Online voting may seem appealing: voting from a computer or smartphone may seem convenient and accessible. However, studies have been inconclusive, showing that online voting may have little to no effect on turnout in practice, and it may even increase disenfranchisement. More importantly: given the current state of computer security, any turnout increase derived from with Internet- or blockchain-based voting would come at the cost of losing meaningful assurance that votes have been counted as they were cast, and not undetectably altered or discarded. This state of affairs will continue as long as standard tactics such as malware, zero days, and denial-of-service attacks continue to be effective.

This article analyzes and systematizes prior research on the security risks of online and electronic voting, and show that these risks not only persist in blockchain-based voting systems, but blockchains may introduce *additional* problems for voting systems. Finally, we suggest questions for critically assessing security risks of new voting system proposals.

DRAFT

1 Introduction

Computers and the Internet have brought great benefits: improving efficiency, reliability, scalability, and convenience of many aspects of daily life. Some naturally ask, *"why don't we vote online?"* Voting online seems tantalizingly convenient: just a few taps on a phone from anywhere, without breaking your daily routine, taking off from work, or waiting in line. However, voting online has a fatal flaw.

Online voting systems are vulnerable to *serious failures*: attacks that are larger scale, harder to detect, and easier to execute than analogous attacks against paper-ballot-based voting systems. Furthermore, online voting systems will suffer from such vulnerabilities for the foreseeable future given the state of computer security and the high stakes in political elections.

While convenience and efficiency are essential properties of election systems, just as security is, these goals must be balanced and optimized together. An election system is ineffective if any one of these goals is compromised.

Exposing our election systems to such serious failures is too high a price to pay for the convenience of voting from our phones. *What good is it to vote conveniently on your phone if you obtain little or no assurance that your vote will be counted correctly, or at all?*

*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).

Those who favor increasing turnout, reducing fraud, or combating disenfranchisement should oppose online voting because the possibility for serious failure undermines these goals. Increased turnout only matters in a system that meaningfully assures that votes are counted as cast. The increased potential for large-scale, hard-to-detect attacks against online voting systems means increased potential for undetected fraud, coercion, and sophisticated vote tampering or vote suppression targeting specific voter groups.

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 [35]) to finding that online voting slightly decreases turnout (e.g., Belgium [21]) to finding that online voting slightly increases turnout but is nonetheless "unlikely to solve the low turnout crisis" (e.g., Canada [37]).[1] Studies of Estonian elections have also suggested that turnout changes due to online voting may favor higher-income and higher-education demographics [74]. Recent U.S. studies demonstrate significant demographic disparities in smartphone ownership (e.g., in gender, income, and education) [66].

Yet proposals for online voting have increased. These proposals are often misperceived as promoting the goals listed above: increasing turnout, reducing fraud, or combating disenfranchisement and coercion. Some online voting proposals have promised added security based on blockchain technology,[2] and have continued development and deployment despite vocal opposition by computer security and blockchain experts (e.g., [47, 48]) and technology reporters (e.g. [8, 39]).

A prominent example is the blockchain-based mobile voting app "Voatz," deployed in 2018 in West Virginia for overseas military voters in the U.S. midterm elections [89, 90], and in several other U.S. states for smaller-scale (municipal/county) elections [54, 73]. Recent research shows that Voatz suffers from serious security vulnerabilities enabling attackers to monitor votes being cast and to change or block ballots at large scale, unnoticed by voters and election officials [77].

[1] See [80] for a concise overview of relevant studies up to 2018, including additional references.

[2] E.g., Voatz, FollowMyVote, and Votem.

A blockchain-based voting system was also used in Moscow, Russia, for its September 2019 city council elections [64]. Though some system code [85] was published and security researchers invited to audit it [50, 63], the system was shown to be gravely vulnerable — not once, but twice (the second time after a proposed fix) [34]. Moscow responded constructively to the first reported vulnerability, but appears to have largely ignored the second. Japan and Switzerland have also conducted smaller blockchain voting experiments [10, 82].

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 [12, 13, 14]). 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 independent 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" [42].

The surprising power of paper A natural but mistaken inclination is to *entirely replace* existing voting methods with the latest digital technologies. Some ask: *"Why wait in polling place lines to cast votes on clunky old voting machines, when votes could be cast from voters' computers and phones over the Internet — using the same security protocols protecting online shopping, banking, cryptocurrency transactions?"*

But, perhaps counterintuitively, getting rid of not only outdated voting equipment but also paper ballots risks "throwing the baby out with the bathwa-

DRAFT

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

ter" and making elections much less secure.

Security considerations for online shopping and online banking are different than those for election systems, in two key ways.

First, online shopping and banking systems have higher tolerance for failure — and *they do fail*. Credit card fraud happens, identity theft happens [84], and sensitive personal data is massively breached (e.g., the 2017 Equifax breach [23]). Online shopping and banking are designed to tolerate failure: merchants, banks, and insurers absorb the risk because doing so is in their economic interest.

Governments may also provide legal recourse for victims (as for the Equifax settlement [24]). But for elections there can be no insurance or recourse against a failure of democracy: there is no means to "make voters whole again" after a compromised election.

Users of Bitcoin and other cryptocurrencies have lost hundreds of millions of dollars [75] due to theft, fraud, or mistake. Cryptocurrencies have fewer risk-absorption mechanisms than traditional banking; losses often fall directly on the victims, with no third party to provide relief.

The second key way in which the threat profile of online banking, shopping, and cryptocurrencies differs from that of elections is the skill level and aims of the adversary. Elections are high-value targets for sophisticated (nation-state) attackers, whose objective is not fraudulent financial transactions but changing or undermining confidence in election outcomes. A technically unsophisticated voter may be attacked by the world's most sophisticated adversaries.

From a computer security perspective, securing an online voting system is a starkly different — and much harder — problem than securing online shopping or banking system.

Surprisingly, low-tech *paper ballots* may help protect against malfunctions or attacks of higher-tech voting system components (as discussed more in §2).

Software independence Voter-verified paper ballots (or paper cryptographic receipts) are the only known way to achieve *software-independence* in voting systems [67, 68]: the property that *an undetected change or error in a system's software cannot cause an undetectable change in the election outcome.*

Although methods exist for improving the reliability and accuracy of software (e.g., using multiple implementations or formal verification), such techniques aim only to ensure correct *processing* of given input data. While valuable, such methods do not ensure that the *input data* (recorded votes) *are correct in the first place*, i.e., that recorded votes accurately capture voters' intent. Only voters can check that their recorded ballot correctly reflects their intent. But if vote-casting is entirely software-based, a malicious system can fool the voter as to the vote actually recorded (cast).

Software independence is an essential requirement for any voting system in a political election. Democracy — and the consent of the governed — cannot be contingent on whether some software correctly recorded voters' choices.

[3] "DRE" stands for "direct-recording electronic." This includes any machine that records votes only electronically (e.g., many touchscreen voting interfaces).

DRAFT

Categories of voting systems This article suggests four main categories of voting systems, determined by two key system attributes (see Table 1):

1. Are votes cast *in person* at a polling site, or *remotely*?
2. Does the system have *voter-verifiable paper ballots* or are ballots represented in a format that is not verifiable by voters (e.g., purely electronic data)?

"Voter-verifiable" means voters must be able to verify *directly* (i.e., without relying on a computer) that their ballot accurately represents their intended vote.[4] For example, a paper ballot is not voter-verifiable if the voter can never inspect it (as, e.g., if voters were to email their choices to an election official, who then prints out a ballot that the voter never sees).

Not every voting system that uses a phone, the Internet, or blockchain technology necessarily falls 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 creation of a paper ballot,[5] and/or saving a copy of the vote cast by paper ballot in an electronic format (perhaps on a blockchain). This article does *not* oppose the use of technology in the context of in-person voting systems with hand-marked paper ballots.

However, almost all proposals billed as "Internet voting," "mobile voting," or "blockchain voting" involve remote voting over the Internet with electronic-only recording of votes; such schemes all fall in the bottom-right category.

Accordingly, this article uses "Internet voting" and "blockchain voting" to refer to schemes in the bottom-right category only.[6] We consider "blockchain voting" a subcategory of "Internet voting," since all blockchain voting proposals transmit information over the Internet.

The top row and the left column of Table 1 are respectively strongly preferable to the bottom row and the right column in terms of security risk. We consider the top row suitable for political elections, with in-person voting preferable to mail-in voting wherever feasible (as indicated by their graduated green color). Importantly, top-row systems are *software independent*; bottom-row systems are not.

We consider the bottom row *unsuitable for political elections for the foreseeable future*, due to their lack of software independence and the greater risk of compromise compared to corresponding alternatives in the top row. Sections 2–3 explain this heightened risk.

The left column of Table 1 is preferable to the right column, because remote voting systems enable coercion and vote selling. Voters using remote voting system lack the seclusion provided by a physical polling place, so a coercer or vote buyer can look over the shoulder of a voter to confirm that they are voting as instructed (or paid) to.[7] In contrast, if voters are secluded at physical polling sites, coercers or vote buyers cannot know the vote really cast, rendering coercion and vote buying 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., [44, 70, 93]). For example, the SAFE Act [44] 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.[8] 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.

[4] There are valid arguments that voter-verifiability *in theory* isn't sufficient if voters don't verify their ballots *in practice* [6, 79]. We skip discussion of this significant point here, as it isn't germane to our main topic.

[5] 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 [30].

[6] We do not distinguish between "mobile voting" and "Internet voting" more generally; mobile voting transmits information over the Internet, and is a subcategory of Internet voting. We avoid the term "mobile voting" henceforth.

[7] Mitigation proposals (such as allowing voters to submit multiple votes but only counting the last one) may help, but only if the adversary can't monitor the voter until polls close (e.g., because the polls close soon, or because they live together).

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

DRAFT

Scope and terminology This article uses "online voting" and "Internet voting" synonymously, in accordance with popular usage, to refer to any system where voters cast votes via the Internet — including blockchain-based and mobile voting systems. We write "electronic voting" to refer to any system where votes are cast purely electronically (i.e., the bottom row of Table 1).[9] Online voting is a subcategory of electronic voting. Much of our reasoning applies to all electronic voting, while some applies only to online or blockchain voting.

This article focuses on systems for *casting and tallying votes* (the focus of recent online and blockchain-based voting proposals). Internet- or blockchain-based technologies may help with other aspects of elections (e.g., auditing or voter registration), but that is not covered here.

Finally, this article focuses on the heightened security required, and particular threats faced, by political elections. Some elections, such as professional society elections, may have less stringent security requirements.[10] Whether electronic voting is suitable for such applications depends on the circumstances, and is not covered here. "Election" should be read as "political election" henceforth.

Election security premises This article posits a few basic premises, listed next, and explains how serious 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.

Organization §2 defines serious failures, and explains how online voting systems are vulnerable to such failures. §3 discusses blockchains and how they might be used in election systems, noting that blockchains do not mitigate any of the weaknesses inherent to online voting systems (from §2), and may sometimes introduce yet additional weaknesses. §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. §5 discusses other related work. Finally, §6 concludes.

2 Vulnerabilities of electronic voting systems

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

We call these *serious failures*: situations where election results have been changed (whether by simple error or adversarial attack) and the change may be undetectable, or even if detected, be irreparable without running a whole new election.

Merely the fact and public perception that the system is vulnerable to such failures may reduce an elected official's legitimacy and therefore destabilize a democracy. Vulnerability to serious failures thus 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 serious failures. For instance, if an election official may handle ballots in secret, they may undetectably destroy ballots cast against a particular 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. This is why most election authorities employ transparency measures, such as

[9] This may include systems that use paper somewhere: e.g., if votes are cast and stored electronically, but a non-voter-verifiable copy of each electronic vote is printed out during the process.

[10] Also, blockchain protocols and smart contracts may employ "voting" as part of their consensus protocol: such protocols are not designed for, and do not meet the security requirements of, political elections, and they are not covered here.

DRAFT

allowing independent observers (including representatives from either party) to monitor and contest any part of the election process [83].[11] Such monitoring enhances accountability in the presence of an auditable paper trail, but could be meaningless if key parts of the election process are shrouded in the internal operations of computers.

Unfortunately, independent observers and monitors have limited ability to prevent such failures: no group has infinite funds, time, and expertise. While acknowledging such limitations, we identify two categories of "showstopper" vulnerabilities that effectively eliminate election authorities' ability to prevent or remediate serious failures.

1. **Scalable attacks:** If the adversary's cost to tamper with the election is much less than the defender's cost for preventing such attacks, attempts to prevent, remediate, or even discover the failures may be impossible in practice. Scalable "wholesale" attacks affecting election outcomes are much more dangerous than "retail" attacks affecting only a few votes.
2. **Undetectable attacks:** If an attacker can alter the election outcome without any risk of the modification being caught (by voters, election officials, or auditors), the attack becomes impossible to prevent or mitigate.

This section next argues that any online voting system 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. These vulnerabilities follow from online voting systems' lack of software-independence.

2.1 Systems attacks

Device exploitation refers to adversarial attacks modifying a computer's hardware, software, or equipment enabling access to information and/or changing the system's operation.

Attackers have complete control over exploited voting systems and how they interact with the voter, including control over what the voter sees. Attackers may prevent casting votes (potentially stealthily, leading voters to believe they did cast votes), deceive voters about any aspect of the voting process, publicly expose voters' choices, or degrade the experience to deter voters from voting at all.

Exploitation is often imperceptible to users, 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 [76]. Such malware is difficult to detect and, after the fact, may remove itself from the system without leaving a trace.

Worse, *any* communication between a system and the outside world may lead to exploitation: even when a device is not Internet-connected (i.e., is "airgapped"). Malware has been installed on airgapped devices, e.g., via USB and other removable media [27].

Systems attacks are incredibly scalable and cost-effective. Perhaps surprisingly, election-scale attacks may be inexpensive. In 2012, an unpatched "zero day" Android vulnerability cost roughly $60,000 [40]. Conservatively estimating that weaponizing, testing, and leveraging the exploit might increase the cost by two orders of magnitude: $6,000,000. For comparison, the total campaign expenditure for one candidate in the 2016 US Presidential election was roughly $768 million [61]. Compared to the research and development budget of a nation-state's intelligence apparatus, exploit costs would be negligible.

Once prepared, a vulnerability may be used many times, and a single use could affect many votes. Attacking centralized services like voting machine manufacturers or voter registries (as in the 2016 U.S. election [46]) may provide a cost-effective way of affecting many votes via few compromised machines, potentially enabling quietly alteration of an election outcome.

Devices are vulnerable, and digital-only defenses are lacking. Device security relies on *many* different organizations. Voting system flaws might be introduced by the voting software vendor, the hardware vendor, the manufacturer, or any third party that maintains or supplies code for these

[11] Specific examples include [20, 25].

DRAFT

organizations. A voter using a phone to vote depends not only on the phone vendor, but on the hardware companies providing 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. This also raises geopolitical concerns: where are devices manufactured, and who controls the voter's network?

Cryptography does not prevent most systems bugs from being exploitable. Conversely, systems flaws may enable *breaking cryptographic guarantees*. Writing software to implement cryptographic primitives and protocols is difficult and subtle [5]; numerous examples have shown systems flaws can lead to compromise of cryptographic systems [4, 15].

2.2 Attacks on systems used in practice

Researchers have repeatedly shown that polling-place electronic-only voting devices are 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 insecure and easily exploited [29], and more recent analyses have show that such systems have not improved [13].

Internet-connected electronic voting has also been attempted and shown to be equally vulnerable. Analyses have been performed on Internet voting systems in Estonia [78], Washington DC [91], and Switzerland [51], all of which were found to be vulnerable to serious failures.

Alarmingly, there is significant evidence that election systems have been penetrated by foreign adversaries. For example, according to the Mueller Report, the Russian government has infiltrated voter registration databases related to Florida and Illinois [46], and there are indications of similar issues in Georgia [94].

2.3 Mail-in ballots

When a voter cannot otherwise access the polls, election authorities may provide a remote voting solution, e.g., mail-in ballots for overseas military and other absentee voters.

However, the risks discussed in this section strongly favor in such cases (1) limiting remote voting to the settings where there is no feasible alternative, and (2) using mail-in ballots rather than online voting. While mail-in ballots enable vote selling and coercion, 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.

See also [28] to read more on the U.S. legal regime governing absentee ballots, including paper ballot requirements.

2.4 End-to-end verifiable voting

Some promising recent proposals called *end-to-end verifiable* (E2E-V) voting systems [2, 3, 11, 19] use cryptographic techniques and post encrypted ballots on a public bulletin board[12] 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 resolve the major problems described in this section. (Paper seems at a minimum necessary to print receipts in an E2E-V voting system, to give the voter credible evidence of any cheating by the voting system.) Thus, any system that is electronic only, even if end-to-end verifiable, seems unsuitable for political elections in the foreseeable future. The U.S. Vote Foundation has noted the promise of E2E-V methods for improving online voting security, but has issued a detailed report recommending avoiding their

[12] §3 discusses how blockchains could be used to implement a public bulletin board. However, we argue that blockchain technology does not add anything *beyond* a way of implementing a public bulletin board, and as such, does not help solve existing issues that E2E-V voting systems share with online voting systems.

DRAFT

use for online voting unless and until the technology is far more mature and fully tested in pollsite voting [33].

2.5 Importance of transparency

Software is complicated; it is very hard to get it right, and software bugs are commonplace. 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 it. 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.[13] Disclosed-source implementations allow one to gain substantial (though not complete) confidence that the implementation contains no serious bugs or security holes.

Disclosing security-critical system designs for inspection by experts and even "the enemy" has been considered good security practice since the 19th century (Kerckhoffs's Principle [49]). While intuition suggests that a secret system design is harder for an adversary to figure out, the lack of scrutiny makes it easier for security vulnerabilities to remain unnoticed and unaddressed. Moreover, keeping a system design secret is infeasible for systems in widespread use — underscoring the importance of security guarantees that hold even if the design is disclosed. 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. One cannot generally 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 difficult and an area of ongoing research.[14] While transparency (disclosed software and good cryptographic protocol documentation) seems necessary for security, it is by no means sufficient.

[13] Here "open-source" means "disclosed-source," where the source code is open for all to read but changes may be controlled. Wallach [88] gives a detailed discussion of open/disclosed source in voting systems.

3 Blockchains as a ballot box

Some recent proposals claim using blockchain technology adds security to electronic voting [32, 86, 87]. We show that blockchains do not address the issues discussed in §2 and might introduce new problems.

We begin by reviewing blockchain technology (§3.1,§3.2). Those familiar with blockchain technology may 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 serious failures described in §2. §3.4 discusses how blockchain-based electronic voting could create additional problems for election systems. 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* is used, confusingly, 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 use of blockchain by Bitcoin [57].

A blockchain implements what cryptographers call a *public bulletin board*: a linear ordering of data with the following properties. It is *append-only*: data can only be added to the end of the board, and never removed; and 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 end

[14] For example, Fink et al. [31] 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.

DRAFT

of the blockchain, and read the transaction list to learn who owns which bitcoins.

Blockchains have validation rules: by consensus, only data with a certain format may be appended. For example, cryptocurrency transactions transferring money must pass certain validity checks or they will not be appended: the sender must have sufficient funds, and the transaction must demonstrate the sender's authorization to move the funds.

Security is guaranteed only under certain assumptions. In Bitcoin, security only holds if a majority of the mining hash power is honest. In other blockchains, the required assumption might be that at least two-thirds of the participants are honest. If such assumptions are violated, the blockchain might lose its availability, linear ordering, and common prefix guarantees.

3.2 How to achieve a blockchain interface

To achieve the public bulletin board functionality, blockchains typically operate as follows. A network of computers runs a common (public) piece of software to agree on an ordered log of data. Users submit new data with digital signatures, and the software enforces validation rules: e.g., users cannot create new coins outside the specified monetary policy. The software also runs a consensus protocol to agree on the continuing log of data, and links the data together using hashes to prevent (undetected) tampering with past data.

Consensus. Distributed consensus is the problem of many computers agreeing on a single value in the presence of failures. Before Bitcoin, 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 whose security is incentive-based, known as Nakamoto consensus [56]. Bitcoin uses a technique called *proof-of-work* [7, 22] 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 guarantees of Nakamoto consensus hold only if the majority of the mining power behave honestly (i.e., follow the protocol).

Some cryptocurrencies implement 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 security guarantees of these protocols hold only if a certain fraction of stakeholders (i.e., coin owners) behave honestly.

The advent of permissionless protocols has caused many to take a second look at distributed databases, where different database nodes 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 they differ in that the participant set is limited and determined ahead of time (nodes need permission to join the system). These protocols improve fault tolerance, and can even tolerate some fraction of malicious nodes (typically up to a third). Distributed database technology can improve databases' resilience to computer failures; however, we shall see that this does not address the core problems with electronic voting from §2.

Authentication. Users create a digital signature to authorize a transaction to be added to the blockchain, perhaps spending coins. There is no "user identity" in the system beyond the signing key itself, and a user may have many unrelated signing keys. Nodes in the network validate signatures and check that each batch of transactions maintains financial invariants, e.g., the spender must have suf-

DRAFT

ficient funds to spend, and/or coins are created following an agreed-upon schedule. In a blockchain without an associated coin, nodes might validate other application-specific rules.

Smart contracts. Blockchains may support operations more complex than just transferring coins: e.g., coins may be transferred conditionally, using 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; these include applications like prediction markets, games, and marketplaces.

Transaction secrecy. By default, blockchains do not keep transaction details secret: all Bitcoin transactions are public. A key feature of blockchain technology is that transactions are verifiable, and public verifiability seems at odds with secrecy. In 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 price: those without access cannot download and verify the blockchain. In a permissionless blockchain (like Bitcoin), the participant set is unrestricted, so the entire transaction history is public. Some cryptocurrencies use *zero-knowledge proofs* to hide transaction details (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 is true. For example, using a zero-knowledge proof, I could convince you that I know the solution to a specific Sudoku puzzle without revealing the actual answer. Zero-knowledge proofs were invented many decades before blockchain technology [36] and may be useful for electronic voting systems (especially E2E-V systems) though they are not enough alone.

Applications. Blockchains have application beyond cryptocurrencies. For example, IBM uses the Hyperledger Fabric blockchain to record the provenance of food traveling through a supply chain [45]. 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 [26]. Its goal is to "create a secure and permanent digital record of an asset's origin, characteristics, and ownership." 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 best-known (but not first [62]) example of blockchain technology, operates in an adversarial environment: anyone can download the software and join the network, including attackers. The idea behind Bitcoin is that participants sign transactions to indicate authorization to transfer, and are constantly downloading and validating the blockchain to check 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.

These ideas seem as though they might be helpful for electronic voting: e.g., using cryptographic signatures to make forging votes difficult, and using hashing and distributed consensus to maintain a ledger of votes that attackers cannot tamper with unless they co-opt 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 §2. In particular, blockchain voting systems are still vulnerable to serious failures, and *the cryptographic and consensus guarantees of blockchains do not prevent serious failures.*

Significantly, *blockchain systems are not software independent*: voters need software to add to or view the blockchain, and a software bug could undetectably change what a user adds or sees (e.g., showing the user that their vote was cast for a certain candidate when it was in fact not).

Next, we sketch a possible blockchain-based vot-

DRAFT

ing system, and discuss how it fails to address several security issues. This design does not consider every detail of implementing a voting system on a blockchain and is not exhaustive, but it demonstrates issues that would apply to many designs.

Coins as votes. Here is a strawman proposal for "blockchain as a ballot box": The voting authority, which maintains a voter registry, has each registered user create a public/private key pair, and each user sends their public key to the registry. Then, the voter registry spends one coin to each public key. To vote, each user spends their coin to the candidate of their choice. After a period, everyone can look at the blockchain, total up each candidate's coins, 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, enabling coercion and vote-selling.

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. An adversary able to influence network connectivity or to conduct a denial-of-service attack could keep users from voting until after the cutoff. 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 — e.g., if 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 these types of attacks, where their blockchains have been rewritten [52, 59, 60]. Blockchains are often referred to as "immutable," but these attacks show that this is not always true in practice, especially for smaller blockchains.

Fourth, security of this strawman hinges on key management. If a user loses their private key, they 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 happened to cryptocurrency exchanges, which have lost hundreds of millions of dollars worth of cryptocurrency to attackers or through bad key management [9, 53]. Blockchains cannot help if a user's keys are compromised; in fact, blockchain-based systems seem to *require* using public key cryptography. This 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, as blockchains alone do not provide software independence. If a user's voting device (probably a mobile phone) is compromised, so is their vote.

Permissioned blockchains. One might think of using a permissioned blockchain, instead, at least to address 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). In permissioned systems, there are even fewer, more homogenous servers to compromise compared to large public blockchain instances. This enhances the possibility that they could all be compromised, especially if they run on the same operating system or run the same software. Permissioned blockchains also do not address the issues of key management or the security of software and hardware on user devices.

Zero-knowledge proofs for secret ballots. Some cryptocurrency schemes keep transaction contents secret while still allowing public verification of certain financial invariants, getting around the tension (described above) between secrecy and public

DRAFT

verifiability. These schemes use the zero-knowledge proofs mentioned in §3.2. For example, Zerocash [71] and its subsequent implementation in the cryptocurrency Zcash [43] provide *shielded* transactions, which do not reveal amounts, senders, or receivers. Despite this, these transactions' financial invariants are still publicly verifiable, much like public blockchain transactions.

One could use these techniques to modify the strawman to support shielded transactions. While this would mean that transaction data would no longer be *publicly* visible, the resulting scheme would still be far from providing ballot secrecy.[15] First, a digital-only solution does nothing to prevent physical monitoring by coercers or vote buyers. Secondly, zero-knowledge proofs are designed for a setting where the party with secret information *wants* to keep it secret (that's why they're using zero-knowledge proofs) — they do not prevent that party from revealing information voluntarily.

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 change.

3.4 New problems blockchains introduce

Besides all the usual security issues associated with online voting, a blockchain-based voting system introduces new security concerns. Blockchains are designed to be decentralized, run by multiple actors. This means blockchain protocols require governance and coordination, which can inherently be difficult to manage(as exemplified in [17, 41]). Importantly, blockchain technology introduces more *complexity* into software and its management. Distributed consensus protocols and cryptographic systems are difficult to implement correctly [1, 18]. Additional complexity means more likelihood that things will go wrong.

[15] Furthermore, adding zero-knowledge proofs would bring new issues related to the complexity and recency of the technology, which is still in early stages. New bugs are being discovered: e.g., in 2018, a critical bug in Zcash was discovered that allowed undetectably counterfeiting Zcash coins [81]. Moreover, the additional complexity may render the voting system (yet more) opaque to the general public, whereas it is important for democracy for the public to believe in the correctness of election technology — and thus, election results.

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, meaning blockchain systems can be vulnerable for longer periods of time than centralized counterparts. In a critical application like voting, the ability to move quickly to fix bugs may be essential.

Other work has proposed frameworks for determining when an application is a good fit for blockchain technology [69, 72, 92]. Though voting requires auditing, it does not warrant the complexity introduced by a technology like blockchain that requires shared governance and shared operation. Elections are inherently centralized (with a central organization, the government, that is in charge of election procedures, the contests of the election, the eligibility of the candidates, and eligibility to vote).

Despite Bitcoin launching in 2009, it took several years to gain users and for developers to gain experience securing the platform. The technology is still new and under development. Another independent concern with using blockchains for voting is the inadvisability of using new cryptographic protocols for critical infrastructure until they have 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 [65]. Augur has a built-in token called REP.

DRAFT

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

A bulletin-board-like interface combined with encryption for secrecy may be helpful for voting, 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, elections are inherently centralized, and decentralized systems come with many drawbacks, including potential congestion and difficulty in upgrading.

4 Critical Questions

As a short article like this cannot provide a comprehensive guide to all of the issues that might be raised about "voting on the blockchain," this section provides the reader with some questions that should be asked about any online or blockchain proposal.

The questions raised here relate to voting system *security*. 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 often overlap! 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 provide operational assistance in the running of an election)

Security objectives

- What security properties is the system intended to have? For each type of adversary, (e.g., foreign powers, corrupt insiders, ...), what behavior is intended to be prevented?
- 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 in the system design? (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 are 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

DRAFT

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? 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 are 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)? Could the recovery procedure be abused?

Remote voting If voting is done remotely:

- What credentials are required to vote? How do voters obtain those credentials? What happens if credentials are lost/stolen?

Operation

- What instructions are given to voters/election officials/others to manage exceptional/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 U.S. National Academies recently produced an excellent report [58] 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 [38] and The Conversation [48].

A collection of related online resources is available on Duncan Buell's website [16]. Finally, we can't resist mentioning the lovely XKCD comic [55] on blockchain voting!

6 Conclusion

A summary of this article's takeaways follows.

DRAFT

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 more vulnerable to serious 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).

Blockchain voting methods fail to live up to their apparent promise. While they may appear to offer better security for voting, they do not help to solve the major security problems with online voting, and might well make security worse.

7 Acknowledgements

We thank Madars Virza and Danny Weitzner for helpful discussions.

Neha Narula and Sunoo Park are supported by the funders of the MIT Digital Currency Initiative. Ronald L. Rivest has received support from the Center for Science of Information (CSoI), an NSF Science and Technology Center, under grant agreement CCF-0939370. Michael Specter is funded by MIT Internet Policy Research Initiative, And Google's Android Security and PrIvacy REsearch (ASPIRE) Fellowship.

References

[1] Ittai Abraham, Guy Gueta, Dahlia Malkhi, Lorenzo Alvisi, Rama Kotla, and Jean-Philippe Martin. "Revisiting fast practical byzantine fault tolerance". In: *arXiv preprint arXiv:1712.01367* (2017).

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

[3] 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.

[4] 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.

[5] 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.

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

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

[8] 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.

[9] B Barrett. *Hack brief: Hackers stole $40 million from binance cryptocurrency exchange*. Wired. 2019.

[10] 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.

DRAFT

[11] 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.

[12] 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.

[13] 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.

[14] 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.

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

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

[17] Vitalik Buterin. *Onward from the Hard Fork*. July 26. URL: https://blog.ethereum.org/2016/07/26/onward_from_the_hard_fork/.

[18] Christian Cachin and Marko Vukolić. "Blockchain consensus protocols in the wild". In: *arXiv preprint arXiv:1707.01873* (2017).

[19] 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.

[20] City and Department of Elections County of San Francisco. *Observe the Election Process*. https://sfelections.sfgov.org/observe-election-process [https://perma.cc/3X5L-ETRW].

[21] 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.

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

[23] 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.

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

[25] European Commission. *EU Election Missions*. http://ec.europa.eu/info/strategy/relations-non-eu-countries/types-relations-and-partnerships/election-observation/mission-recommendations-repository/home [https://perma.cc/KKL4-EU6N].

[26] *Everledger*. https://www.everledger.io/. Feb. 12, 2020.

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

[28] Federal Voting Assistance Program. *The Uniformed and Overseas Citizens Absentee Voting Act Overview*. https://www.fvap.gov/info/laws/uocava.

[29] 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,*

DRAFT

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.

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

[31] 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.

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

[33] 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.

[34] Pierrick Gaudry and Alexander Golovnev. *Breaking the Encryption Scheme of the Moscow Internet Voting System*. Proc. Financial Cryptography '20. http://fc20.ifca.ai/preproceedings/178.pdf.

[35] 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.

[36] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. "The Knowledge Complexity of Interactive Proof Systems". In: *SIAM J. Comput.* 18.1 (1989), pp. 186–208. DOI: 10.1137/0218012. URL: https://doi.org/10.1137/0218012.

[37] 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.

[38] 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.

[39] 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.

[40] 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).

[41] Colin Harper. *Bitcoin Independence Day: How This Watershed Day Defines Community Consensus*. Bitcoin Magazine. Aug. 1. URL: https://bitcoinmagazine.com/articles/bitcoin-independence-day-how-this-watershed-day-defines-community-consensus.

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

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

[44] *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].

[45] IBM. *IBM Food Trust*. https://www.ibm.com/blockchain/solutions/food-trust. Feb. 12, 2020.

DRAFT

[46] 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.

[47] 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.

[48] 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.

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

[50] 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.

[51] Sarah Jamie Lewis, Olivier Pereira, and Vanessa Teague. *How not to prove your election outcome*. Technical Report. Mar. 2019.

[52] James Lovejoy. *Bitcoin Gold (BTG) was 51% attacked*. Jan. 2020. URL: https://gist.github.com/metalicjames/71321570a105940529e709651d0a9765.

[53] Robert McMillan. *Bitcoin exchange Mt. Gox implodes amid allegations of $350 million hack*. Wired. Feb. 24.

[54] 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.

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

[56] Satoshi Nakamoto et al. *Bitcoin: A peer-to-peer electronic cash system*. 2008.

[57] Arvind Narayanan and Jeremy Clark. "Bitcoin's academic pedigree". In: *Communications of the ACM* 60.12 (2017), pp. 36–45.

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

[59] Mark Nesbitt. *Deep Chain Reorganization Detected on Ethereum Classic (ETC)*. Jan. 2019. URL: https://blog.coinbase.com/ethereum-classic-etc-is-currently-being-51-attacked-33be13ce32de.

[60] Mark Nesbitt. *Vertcoin (VTC) was successfully 51% attacked*. Dec. 2018. URL: https://medium.com/coinmonks/vertcoin-vtc-is-currently-being-51-attacked-53ab633c08a4.

[61] 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).

[62] Daniel Oberhaus. *The World's Oldest Blockchain Has Been Hiding in the New York Times Since 1995*. https://www.vice.com/en_us/article/j5nzx4/what-was-the-first-blockchain. 2018.

[63] 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].

[64] 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].

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

DRAFT

[66] Pew Research Center. *Mobile Fact Sheet*. https://www.pewresearch.org/internet/fact-sheet/mobile [https://perma.cc/9DFC-G3LG]. June 12, 2019.

[67] 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.

[68] 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.

[69] 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.

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

[71] 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.

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

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

[74] 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.

[75] 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.

[76] Sherri Sparks and Jamie Butler. "Shadow Walker: Raising The Bar For Windows Rootkit Detection". In: *Phrack Magazine* 0x0b.0x3d (2005). URL: http://phrack.org/issues/63/8.html.

[77] Michael A. Specter, James Koppel, and Daniel Weitzner. *The Ballot is Busted Before the Blockchain: A Security Analysis of Voatz, the First Internet Voting Application Used in U.S. Federal Elections*. Preprint available at: https://internetpolicy.mit.edu/wp-content/uploads/2020/02/SecurityAnalysisOfVoatz_Public.pdf. 2020.

[78] 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.

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

[80] 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.

[81] 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/}.

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

DRAFT

_ - switzerland - s - first - municipal - blockchain - vote - hailed - a - success / 44230928. 2018.

[83] S.W.L. *What do election observers do?* The Economist. https://www.economist.com/the-economist-explains/2017/06/21/what-do-election-observers-do [https://perma.cc/XHV5-SWHG].

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

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

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

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

[88] 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.

[89] 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.

[90] 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.

[91] 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.

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

[93] *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.

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

DRAFT