Cybersecurity Considerations for Voting Systems

Wenke Lee, Ph.D.
wenke.lee@gmail.com

This document provides a very brief overview of cybersecurity and discusses the design considerations for secure voting.

As presented to the Georgia Office of the Secretary of State's "Secure, Accessible & Fair Elections (SAFE) Commission," on Aug. 30, 2018.

Wenke Lee

- Work at Georgia Tech (2001-)
  - Professor of Computer Science, John P. Imlay Jr. Chair
  - Co-Executive-Director of the Institute for Information Security & Privacy (IISP)
  - Teach cybersecurity to 2,500 students/year
- Researcher in cybersecurity (1994-)
  - Ph.D. in 1999 from Columbia University (Thesis: a machine learning framework for intrusion detection)
  - Systems and network security, malware analysis, botnet detection, cryptography; Damballa (Core Security)

My name is Wenke Lee and I am a professor and John P. Imlay Jr. Chair in the School of Computer Science, College of Computing, at the Georgia Institute of Technology.

I am also a co-executive director of the Institute for Information Security & Privacy (IISP) at Georgia Tech -- the coordinating body for 13 cybersecurity research labs at Georgia Tech which together performed more than $144 million in research for government, defense, and corporate partners last year (FY2018, ended June 30, 2018). The IISP's mission is to unify research scientists, faculty and students across multiple fields – such as Engineering, Business, and Public Policy – to support comprehensive research, new educational pathways, and the tech-transfer that moves our discoveries out of the university and into the marketplace.

I have been a researcher in cybersecurity for more than 20 years. I received my Ph.D. in Computer Science from Columbia University in 1999. For my thesis research, I developed a machine learning framework for intrusion detection.

I have been a professor since 1999. Today, I teach several cybersecurity classes at Georgia Tech to approximately 2,500 on-campus and online degree students per year.

I also continue to perform cybersecurity research for partners such as DARPA, the Office of Naval Research Labs, National Science Foundation, Intel and others. I have published over 100 peer-reviewed papers in top academic venues about systems, network, and software security; malware analysis; botnet detection; authentication, and data encryption. Some of my research about botnet detection was used to start an Atlanta-based company, called Damballa, which was later acquired by Core Security.

SAFE Commission

- Work and opinion: my own
- Input from researchers in voting system security

The security and integrity of our voting system is essential for our democracy, and so I am truly honored to serve on the SAFE Commission. I have been a citizen of the United States for nearly 20 years. I deeply appreciate that within a democracy, you can advocate for yourself and for others, that you can shine light on policies that need improvement, and openly discuss better approaches to self government. The choices we make on election day are central to this process and the election process must be protected in every way .

My work for the SAFE Commission is my own and therefore my opinions do not necessarily reflect those of Georgia Tech.

I rely on my decades of experience in cybersecurity, as well as input from computer science and engineering researchers working in the area of voting system security. I have been reading their papers and reports, and I have had direct discussions with them about various voting security issues.

Vulnerabilities In Voting Systems

- There will always be … not news!
- Vulnerabilities = errors/weaknesses that can be exploited by attackers

- No system can be shown to contain no error
  - Developed by engineers/programmers
    - Chrome: 7 million lines of code, Android: 15 million Windows: 50 million, Car: 100 million+
  - *Can you write/edit a book that thick without an error?*

We often hear in the news that a cyberattack or a hack occurred, and unfortunately, lately some of those news stories involve voting systems. Typically, a cyberattack occurred because an attacker was able to exploit a vulnerability in a cyber system.

I am never surprised when I hear news about any cyberattack. There is an established theorem that states, “there is no way to know for sure that any real, useful system contains no vulnerability.” That is, even if we carefully engineer and test a system, we still cannot be sure that it has no vulnerability (or no error); and much more likely than not, any system will have some security vulnerabilities.

This should not be a surprise given how complex today’s systems are: for example, Chrome has 7 million line of code, Android has 15 million, and Windows has 50 million, and a typical automobile control system has 100 million lines of code. Cyber systems are developed and tested by programmers and engineers, and so errors introduced by humans are not avoidable.

EVERY System Is At Risk

- Not if but when, how much can we find out?
- Even sophisticated, high-profile organizations have been attacked
  - e.g., OPM, HomeDepot, Equifax
- Many organizations seek public help to secure their systems
  - DoD, Google, Apple, Tesla, United Airlines

We often say that the question is not IF an organization will be hacked; it is WHEN, and how much can we find out about the damage afterward. This is because every system is likely to have security vulnerabilities.

Even high-profile organizations, such as HomeDepot and Equifax, can be vulnerable despite investing a lot of resources in security protection. Organizations continually must go to great lengths to improve their security protection.

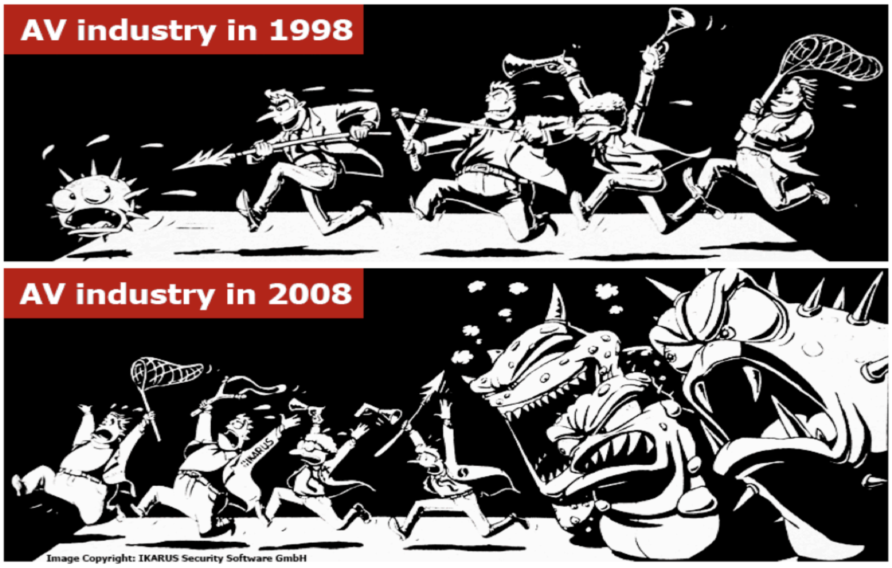
It is commonplace now for companies to offer “bug bounties” -- financial rewards to anyone who openly hacks, finds, and discloses security vulnerabilities in their products, services, and operations. Tech giants, airlines, even the U.S. Marine Corps have publicly advertised and invited hackers to comb their systems for flaws **and report them.**

These types of hacks are invited by the organization, seen as helpful to the organization, and performed by “white-hat hackers” who are acting for the public good. Within cybersecurity circles, we always say it is better that a White Hat or academic researcher find your flaws, because we report them when we do. The bad guys don’t.

The reporting of flaws can be made directly to the organization, to a vendor responsible for the flaw, and/or to a federal organization such as the U.S. Department of Homeland Security CERT team – the Computer Emergency Readiness Team.

The point here is that everyone is vulnerable, everything is at risk, and it takes a community to help counter cyberthreats.

Cybersecurity Is VERY Hard



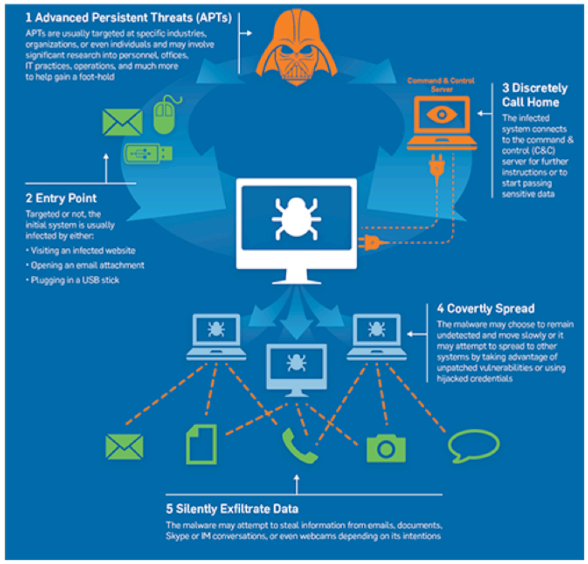
Why can't we just secure our cyber systems?

First, in order to do so, we need to know what vulnerabilities our systems have and fix them all. But if that is achievable, it would mean we have found a way to show that a (fixed) system has no vulnerability, which is a contrary to the established theorem. In short, **we can never know that we have completely secured any system.**

Second, attacking is inherently easier than defending because it needs to only find and exploit the weakest link in a system or organization. And attack technologies have advanced significantly since the late 1990s to become easily accessible to any would-be attacker. For example, an attacker can "buy" and deploy malware that exploits a previously untapped vulnerability to compromise a large number of computers or just target a specific organization.

Or he can simply "rent" the right to use a list of already compromised computers, known as "bots" which have a vulnerability that has not been patched. Simply put, attacks have become easier and hence more prevalent.

Advanced Persistent Threats



A particularly dangerous kind of attack is the so-called "Advanced Persistent Threat" (APT). Such an attack aims to quietly carry out its malicious activities so that it is hard to not only detect the attack but also access the damage upon detection.

An APT attack starts by compromising a computer (or a user account) via many means, including phishing emails, compromised web sites, "free" USB thumb drives, etc. Once it succeeds in injecting a malware to run on a compromised computer, the malware connects to the attacker's computer to receive commands and updates, and accordingly carry out the intended attack, such as spreading the malware to other vulnerable computers or user accounts in the organization and exfiltrating any valuable data to the attacker's computer. APT malware is designed to carry out activities below the detection threshold, e.g., transmitting data in small volume or only when the user is also browsing the web, and even removing all evidences to cover its tracks as it moves from one computer to the next.

Achieving Cybersecurity

- Secure = not vulnerable to cyber attacks
- Option #1: Don't use any cyber component
  – Because we can't guarantee zero vulnerabilities
- Option #2: Keep away would-be attackers
  – But the cyber world is VERY connected
    - e.g., from Internet-facing systems to "disconnected" systems via media (e.g., Stuxnet)
  – "Insider" attacks
    - Compromised account = insider

What can we do to achieve cybersecurity? That is, how do we ensure our mission is not vulnerable to cyberattacks?

The first, obvious approach is to not use any cyber system in our mission because we cannot be sure that a cyber system is ever truly protected from attacks. By eliminating all cyber systems, by definition, our mission cannot be compromised by cyberattacks. On the other hand, this is not always the best approach because cyber systems provide many profits, e.g., automation for efficiency, accessibility, etc.

The second approach is to keep our cyber systems away from the would-be attackers so that they cannot compromise our systems. This is very hard to achieve for two reasons. First, the cyber world is very connected, often in ways that are surprising to users and system administrators, and therefore it is very hard to keep outside attackers from reaching into an internal system. For example, we could disconnect a system from the Internet by not allowing any network connection, but there may still be an indirect way that the system interacts with the Internet, e.g., if the user plugs in a USB thumb drive with data from another computer that was connected to the Internet. The Stuxnet malware that attacked Iran's nuclear capabilities indeed used this method to infect controllers that were not directly connected to the Internet.

Finally, would-be attackers are not necessarily always outside of our organization. There is always the possibility of an "insider" attack by a rogue staff member or volunteer. In addition, if an attacker has already compromised a user account, then he becomes an "insider" because he can now log in as the user.

Achieving Cybersecurity

- Option #3: Be practical (not absolute 0 or 1)
  - *Security* is a state of well-being for information and infrastructures in which the possibility of successful yet *undetected* theft, tampering, and disruption of information and services is kept low or *tolerable*
    - Confidentiality, authenticity, integrity, availability

The practical approach to achieving cybersecurity is to define security not in absolute terms. Instead, we should use the following definition (in fact this is the textbook definition I use in my classes):

Security is a state of well-being for information and infrastructures in which the possibility of successful yet undetected theft, tampering, and disruption of information and services is kept low or *tolerable*. Security goals include: confidentiality, authenticity, integrity, and availability.

That is, to secure our system means we need to keep the possibility of successful and undetected attacks sufficiently low for our critical missions. In other words, security is about understanding and mitigating risks.

Achieving Cybersecurity

- The security life cycle
  - *Iterations* of
    - Threat and risk analysis
    - Policy decisions
    - Specification
    - Design
    - Implementation
    - Operation and maintenance

Much like how one maintains good health by constantly repeating good habits, achieving cybersecurity requires the constant practice of security process and measures. More specifically, we need to keep iterating the following steps:

- **Threat and risk analysis** – Identify the valuable assets that may be targets of attackers. Identify (new) potential attackers, their motivations, targets and methods. Analyze the likelihood that particular attack will succeed and go undetected.
- **Policy decisions** – Decide the most important assets that we must protect from cyberattacks. That is, decide what risks we cannot tolerate, and what risks we can accept.
- **Specification** – According to the policy decisions, specify which security features are needed.
- **Design** – According to the specification, determine the necessary functionalities of technology components and how they should work together.
- **Implementation** – According to the design, construct the system and test it to verify that it provides the specified and desired features.
- **Operation and maintenance** – Deploy and operate the system according to its intended functions and apply up-to-date patches. Human operators must be trained to properly understand and use the system.

Cybersecurity In Voting Systems

- Threat and risk analysis
  - "Rank order" threats based on
    - Impact, success probability, attribution potential
    - Can a remote attacker change MANY votes?
      - And what are the components that can be targeted?
    - Can a few attackers with access (e.g., posed as worker or voter) change MANY votes?
    - Can a remote attacker shutdown (i.e., make unavailable) the key components (e.g., reporting)?
    - Etc.

Let's discuss the main cybersecurity issues in voting system.

We should follow the security lifecycle, and the first step is threat and risk analysis. We can analyze the threats according to the potential impacts of an attack, its success probability, and our ability to attribute the attack (attribution is a deterrent to a would-be attacker and can reduce the likelihood of an attack).

For example, since the most important "asset" of a voting system is the vote and attackers will attempt to change votes, we can consider:

1. Can a remote attacker – not at the polling station – change MANY votes?

    And what are the components that can be targeted in order to do so?

    This is the most devastating attack because of the potentially large impact and the difficulty of identifying a remote attacker.

2. Can a few attackers with access (e.g., posing as a worker or voter) change MANY votes?

    This attack also can have a large impact, but since it requires physical access, it is more cumbersome for an attacker.

3. We should also consider the availability of the voting and election systems because it affects voter turn-out and is therefore a potential attack target:

    Can a remote attacker shutdown (i.e., make unavailable) the key components of the system (e.g., tabulation and reporting)?

Cybersecurity In Voting Systems

- Policy decisions
  - What is really important? Or, what risks can we tolerate (and to what extent) and what can't we?
  - Integrity: votes are accurately counted
  - Voter confidence:
    - Verifiably cast-as-intended
    - Verifiably collected-as-cast
    - Verifiably counted-as-collected
    - *Any* cyberattack can erode voter confidence

The most critical cybersecurity risk in a voting system is that votes are not counted accurately as a result of cyberattacks. Any successful cyberattack, or the belief that a successful attack is inevitable, will erode voter confidence and inflict great harm to our democracy.

All voters deserve to be confident that their votes are counted correctly. For a voter's vote to be counted accurately by the voting system, we need to ensure that the vote is cast in the voting system as intended by the voter, is collected by the voting system as cast, and is counted by the voting system as collected.

Cybersecurity In Voting Systems

- Specification and Design
  - Strong software independent
    - An undetected change or error (including cyberattack) in software cannot cause an *undetectable* change or error in an election outcome; and
    - A detected change or error (due to software) can be corrected without rerunning the election
    - Can recover from cyberattack but requires other trail of evidence (that cannot be affected by the software)

How do we ensure that votes are counted accurately when cyber component(s) are used in the process and we already know that they very likely have security vulnerabilities?

One security feature that a voting system must have is to be “strong software independent.” That is:

- an undetected change or error (including cyberattack) in software cannot cause an undetectable change or error in an election outcome; and
- a detected change or error (due to software) can be corrected without re-running the election.

If a voting system is strong software independent, then it can recover from cyberattacks, but this obviously requires another trail of voter evidence that cannot be tampered or deleted by the software.

Cybersecurity In Voting Systems

- Specification and Design
  - Paper ballots
    - (If done right) Durable evidence to determine correct election outcome
      - Must secure the custody of paper ballots
  - Statistics and auditing
    - Continue to examine random samples of ballots, until
      - There is strong statistical evidence that the election outcome is correct, or
      - There has been a complete manual tally

How do we design a voting system that is strong software independent? We need to maintain voter evidence that cannot be affected by software.

The best approach is to use paper ballots as the durable, independent evidence to verify or determine the correct election outcome, assuming that the paper ballots have accurately captured the voters' intended votes. Obviously, we must also secure the custody of the paper ballots.

With paper ballots, we can:

- apply risk-limited-auditing to verify or determine the correct election outcome;
- continue to examine random samples of ballots and manually count the votes until there is strong statistical evidence that the election outcome is correct, (i.e., the results of manual counting agree with the results of a tallying cyber system), or there has been a complete manual tally. In this case, the tallying cyber system must have functioned improperly either due to a cyberattack or some other error, and we just use the result of the complete manual tally as the correct election outcome.

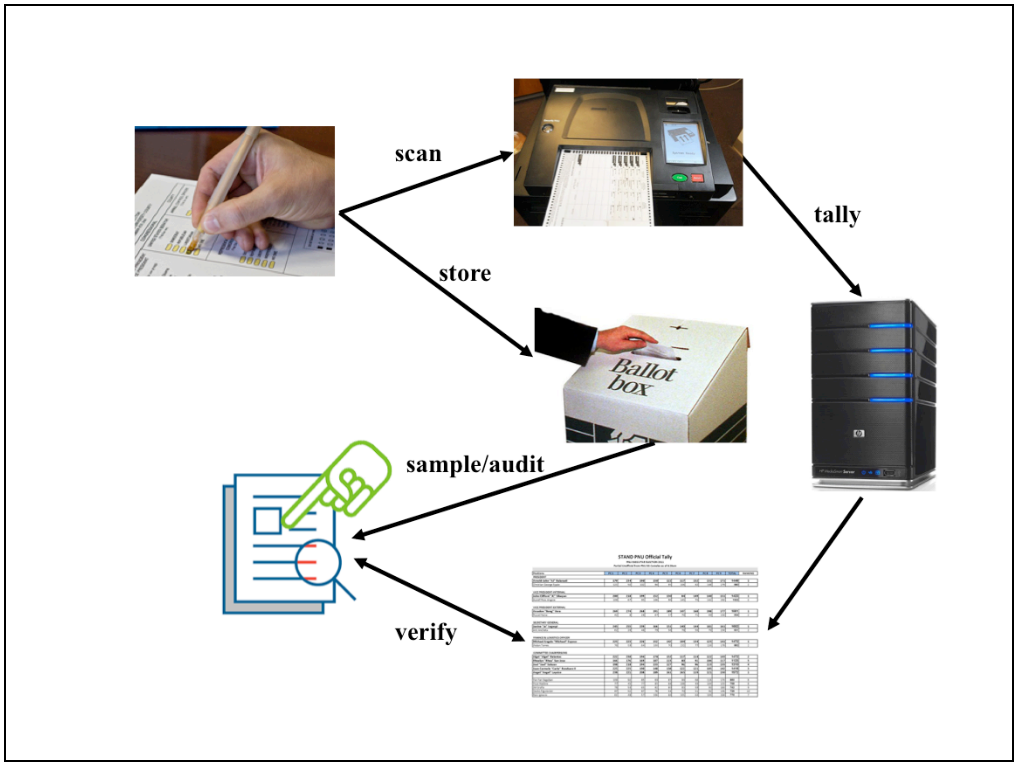
Cybersecurity In Voting Systems

- Specification and Design
  - Paper ballots done right - with auditing, will accomplish:
    - Verifiably cast-as-intended
    - Verifiably collected-as-cast
    - Verifiably counted-as-collected
    - Cannot be completely controlled/manipulated by any cyber component
      - Voters commit/verify votes on ballots
      - Readable/countable by human

How do we ensure that the paper ballots accurately capture voter intent and can be used reliably in an audit?

First, we need to ensure that the voters commit and verify their actual votes on the ballots. **The best approach is to require the voters to hand mark paper ballots that are then scanned and tallied by cyber systems but also dropped in a safe box.** This is because marking each vote captures and verifies the voter's intention in a single act. The much less desirable approach is to have a voter cast his vote on a ballot-marking device, with a cyber component, and print out a paper receipt that the voter would verify and also drop in a safe box. This approach is not secure because the ballot-marking device may have a vulnerability that can be exploited to change votes. Asking the voter to simply read a print-out receipt as verification of his action is an additional step that can be simply ignored by the voter. The difference between these two approaches is critical: with hand-marked paper ballots, a voter both casts and verifies (that is, he verifies as he marks and he cannot cast without already verifying); but with ballot marking devices, he can skip the verification step.

Second, regardless of whether a paper ballot is hand-marked or is a print-out receipt from a ballot-marking device, it must be easily and clearly readable and manually countable. In particular, it must show each and every vote exactly as the voter cast it. It cannot be just a summary of the votes (e.g., that is only a tally, or shows the presidential ballot and omits down ballots). It absolutely cannot be a barcode, QR code, or any other kind of encoding scheme that is readable only by a machine because the cyber system that reads the ballots also can be compromised and lie to the voter or auditor. During a manual review, a human must be able to view evidence of the voter's original act.



This figure illustrates the workflow and summaries the design of a voting system that is strong software independent, that is, a system that can recover from any cyberattack without the need to re-run an entire election. This is the “gold standard” for voting systems from the point-of-view of cybersecurity researchers and computer scientists who have studied election systems.

A voter is given a paper ballot, he marks the intended vote, then he puts the ballot on a scanner to have the machine record the vote, and once the scanning is done, the voter also drops the ballot in a safe box. The scanning machine forwards the recorded votes to a tallying machine, which counts the votes from all voters and outputs the election result. Auditors may then open the safe box to perform a risk-limiting audit, (i.e., manually read and count samples to verify that the outputs from the tallying system are correct).

Cybersecurity In Voting Systems

- Implementation
  - Not all cyber systems are created equal
    - Choice of hardware and operating system
    - Choice of programming language
    - Secure coding practice
    - Review (open design/source)
    - Penetration testing, bug bounty
  - Latest security technologies
    - E.g., new hardware and software components specifically designed to provide security protection
    - *Don't use the same system for more than a few years!*

There are many approaches to implement the same design, and cybersecurity should be the first consideration when making decisions. Ideally, we should use:

- hardware that provides built-in security support, such as a cryptography engine, trusted platform module, etc.
- operating system that provides the latest security technologies, such as sandboxing, application signing, etc.
- programming language that reduces the chance of programming errors (e.g., buffer overflow) that can lead to security attacks.
- secure coding practice that emphasizes correctness and safety (e.g., always performing bounds check) over efficiency.
- review of design and code that checks security vulnerabilities. When possible, use an open-source system (or components) that can be reviewed by many experts.
- penetration testing to identify potential attacks and, when possible, establish a bug-bounty program to invite experts to test the system.

Technology vendors are always hard at work developing new technologies that provide better security protection. For example, in the past five years, there have been major, generational advances -- not mere updates -- in hardware (with a built-in cryptography engine) and operating systems (with mandatory application signing) that enable systems that utilize these new technologies to be significantly more secure.

Given the importance of cybersecurity, we want our voting systems to be built on top of the latest generations of hardware and operating-system technologies. That is, we should be using the same systems for **no more than five years.**

Cybersecurity In Voting Systems

- Operation and maintenance
  - Adopt best practices in cybersecurity, e.g.,
    - Strong authentication and data encryption
    - Blocking and detecting bad activities at network perimeters as well as endpoints
    - Up-to-date security patches
    - Penetration testing
    - Training, e.g., anti-phishing

Cybersecurity requires constant vigilance at all components and by all parties. That is, we need to always use the best practices:

- Enforce strong authentication, such as two-factor authentication
- Encrypt whenever possible, that is, encrypt all data at rest and all network traffic that does not need to be in the clear.
- Protect both the network perimeter and endpoints, that is, use intrusion prevention and detection systems to block and detect bad activities to the network as well as on endpoint computers.
- Diligently apply up-to-date security patches, in fact, all systems should be set to automatically download and apply security updates.
- Schedule regular penetration testing by third-party providers and make improvements according to findings.
- Perform regular user training (e.g., use training tools to teach users how to identify phishing emails).

In Summary

- Any cyber system is vulnerable.
- "Strong software independent" systems require a human-verifable element to recover from cyberattack and retain voter confidence.
- Paper ballots are the durable, independent trail of voter intent.
  - Voters hand-mark their paper ballots, submit the paper ballots to the scanning machine, and drop them in a safe box.

Any cyber system is likely to be vulnerable to attacks. For voting systems, a cyberattack can potentially change a very large number of votes and hence the outcome of an election. More importantly, any cyberattack on voting systems, regardless of its real impact, will severely erode voter confidence and affect future voter participation.

Therefore, we need a voting system that can recover from any cyberattack without the need to rerun the election. That is, such a system will give voters the confidence that their votes will never be compromised by cyberattacks. This can be achieved by making the voting system "strong software independent," which in turn requires paper ballots as the durable, independent trail of voter intent that can be manually audited by humans (through sampling and counting). The gold standard is to have the voters hand-mark their paper ballots, submit the paper ballots to the scanning machine, and once scanned drop them in a safe box. This approach guarantees that the voters verify their intended votes while casting the votes, and the risk-limiting auditing process will guarantee that the votes are collected and counted accurately; that is, this gold-standard approach guarantees that votes by the voters are counted accurately.

Acknowledgement

In this report, I have borrowed important concepts, in particular, "strong software independent," from the report "Public Evidence from Secret Ballots" (see https://arxiv.org/abs/1707.08619). One of the primary authors is Professor Ron Rivest at the Massachusetts Institute of Technology and the "R" of RSA, the most widely-used public-key cryptography algorithm. Professor Rivest is also a winner of the Turing Award, internationally recognized as "Nobel Prize" for computer science.