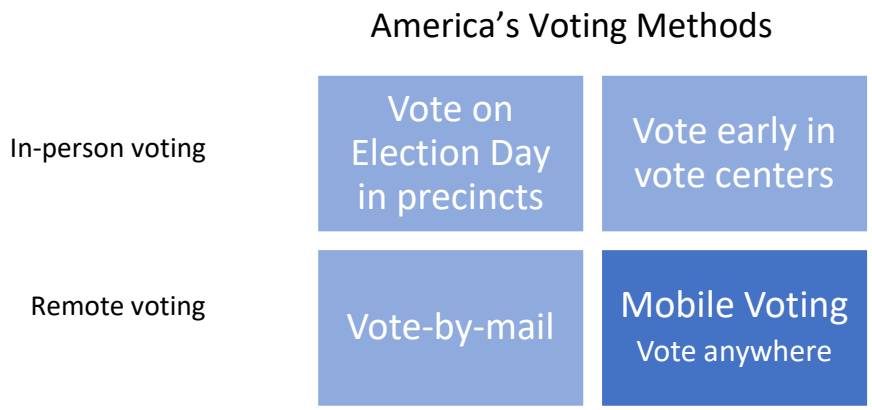
Voatz Mobile Voting Platform

Security, Identity, Auditability: An Overview

Version 1.0

Abstract

Mobile voting is a much-needed, long overdue response to the challenges faced by an important segment of the American electorate: people whose circumstances make it impossible or inconvenient to vote conventionally. The Voatz Mobile Voting Platform adds a new, fourth voting method to the ways America's electorate votes, as shown here.



This paper introduces security concepts and technologies that are widely used in other industries. It describes how Voatz has incorporated them into a new, resilient remote ballot marking system that addresses the three major security-related obstacles that have stymied progress in online voting:

- **Security: Device security, network security and secure storage of returned ballots**

  How can we be reasonably confident that a) the voter is using a device that has not been compromised with malware; b) network-based attacks can be detected and mitigated; and c) the storage of returned voted ballots be made tamper-proof? Additionally, since "security" is an ongoing journey, how can we recover from an unforeseen threat?

- **Identity: Identity proofing, voter authentication and binding**

  Ahead of the election, how can we be reasonably certain that the remote voter presents valid, unexpired government issued identification, and is the same person on the remotely presented credential? During the voting window, how can we be reasonably certain that the user is the same person who previously established their identity, that they are registered to vote, and that the principle of one person, one vote is preserved?

- **Auditability: Voter-verified and transparent, jurisdiction run post-election audits**

  How can the voter and the public trust that voter intent was accurately recorded, transmitted and counted?

Although not specifically security related, **usability and accessibility** are critical to the success of a system in the field even if it adequately addresses the issues of security, identity and auditability.

> **[In 2015,] it is currently unclear whether it is possible to construct an E2E-VIV system that fulfills the set of requirements contained in this report. Solving the remaining challenges, however, would have enormous impact on the world.**
>
> **The Future of Voting: End-to-End Verifiable Internet Voting**[i]
>
> **2015, U.S. Vote Foundation and Galois, Inc. with financing from the Democracy Fund**

Contents

Voatz Mobile Voting Platform Whitepaper: Security, Identity, Auditability

Overview

The Voatz Mobile Voting Platform was created by applying security and smartphone expertise to the experience of building and certifying a primary voting system. The team developed their knowledge of security and smartphone applications by designing and programming highly secure mobile payments applications for major financial institutions. The team's election expertise was gained over ten years during which members of Voatz designed and built a federally certified voting system. The combination of mobile security expertise and practical voting system expertise is a unique combination of skills in the election industry. As entrepreneurs, they identified and forged relationships with other companies that contributed tools and technologies already in widespread use in other industries. Together, they were able to address the persistent issues that have stymied progress to create a more secure, convenient, accessible and resilient voting experience, and provide election administration a more economical and responsive approach to reach the most challenging segment of the electorate – citizens whose circumstances make it difficult for them to vote conventionally.

Problems to be solved in mobile voting

Over the course of their extensive elections research, the Voatz team realized that the current approaches, where security features were "bolted onto" legacy software, would not work. Only through a comprehensive approach to security and accessibility would it be possible to enable voters, regardless of their circumstances, to vote securely from virtually anywhere in the world.

There is general agreement[ii] on the obstacles that have prevented progress towards Internet-based remote voting. These obstacles can be grouped into three categories:

Security: Device security, network security and secure storage of returned ballots – The ability to detect that the voter's smartphone has not been critically compromised and, if compromised, prevent the voter from accessing or submitting a ballot. In addition to the detection of malware, included are the methods to protect against denial of service (DoS) attacks and the safe return and storage of voted ballots.

Identity: Identity proofing, voter authentication and binding – *Identity proofing* is the ability to be reasonably certain that a government-issued credential is valid and that the person presenting their credential, who can be located virtually anywhere, is who they say they are. When the time comes to vote, *authentication* verifies that the same person whose credentials were validated is the same person attempting to vote. Finally, the concept of *binding* provides reasonable certainty that a registered voter can only vote on one device and that that voter cannot vote on another device.

Auditability: Voter-verified and transparent, jurisdiction run post-election audits – Like in-person voting, voters should have the ability to verify that their initial selections were recorded as intended, given the opportunity to spoil their ballot and vote a new one, and then verify that their selections were properly rendered for tabulation. The jurisdiction should also be able to verify voter intent and that the primary voting system accurately reported results from ballots cast remotely.

A note on absolute anonymity

Voatz recognizes that it would be desirable to guarantee absolute voter anonymity of ballots submitted remotely. However, given that every state needs to know *who has voted* (a requirement called "voter credit"), and given that nearly every state requires the remote voter to waive their right to anonymity, Voatz can make it extremely difficult, *but not always impossible,* for the election department staff to pair

a voter's ballot with their identity, especially when very small groups of voters are involved. Also, Voatz recognizes that if a remote voter can demonstrate how they voted, there will be concerns over the potential for coercion and vote buying. However, given that most states already send ballots through the postal mail, there is no practical technology that can prevent these violations. Legislation and enforcement are the best practical methods of deterring coercion and vote buying.

Pilots: The Key to Innovations in Elections

Every election official knows elections are complex. That is why the successful introduction of a new technology is always preceded by small, well-designed pilots administered by responsible election officials. In fact, since scalability can largely be simulated in a lab, having many participating voters at the outset is generally not desirable. Rather, it is more important to address the issues that cannot be simulated, which include:

- **Feedback from as diverse a set of voters as possible.**
- **Jurisdiction-specific workflows,** which include how voters request a mobile ballot, how they are notified of their eligibility to vote on their smartphone, how registration status is confirmed, and how exceptions, such as requests to spoil a ballot and signature match failures, are handled.
- **Compliance** with statutory requirements, administrative rules and jurisdictional procedures.

Elections are not academically designed "controlled experiments," which can be performed repeatedly with measurable outcomes. Elections are infrequent and requirements vary across elections (e.g., between primaries and general elections) and between states (e.g., open versus closed primaries).

Numerous, small pilots are virtually the only way technology providers can learn how to ensure that a) their product fits with the requirements of a jurisdiction; b) voters of all abilities enjoy their experience; and c) non-technical people can trust that voter intent was properly recorded, safely transmitted, securely stored and accurately tabulated.

The remainder of this paper introduces the methods Voatz uses to ensure security, voter identity and auditability. This paper is meant for a curious, non-technical reader who wishes to familiarize themselves with the various concepts that comprise a modern online voting experience.

Security

The topic of security is broad. To make it easy to digest, the following section divides the topic into three categories: device security, network security and the secure storage of voted ballots.

Device Security

Mobile Threat Defense (MTD) is a growing segment in smartphone security and is analogous to PC-based virus and malware detection. Voatz integrates licensed technology from Zimperium, an industry-leading provider of MTD services. When the voting application is launched, MDT software embedded in the Voatz application automatically performs three types of tests which include:

> ***Device vulnerability assessment*** – MDT tools inspect the smartphone for configuration weaknesses such as a "jailbroken" (iOS) or "rooted" (Android) phone or unnecessarily elevated privileges that could lead to malware execution. Upon detection of a critical threat, Voatz will not allow a voter to request a ballot.
>
> ***Application scans*** –MDT checks for applications on the iOS or Android smartphone that have not been digitally signed by either Apple or Google respectively.
>
> ***Network security assessment*** – MDT tools monitor network traffic and flag suspicious connections, like man-in-the-middle attacks (MITM).

Detection of critical threats will prevent the voter from beginning ID verification or submitting their voted ballot. In addition to critical threats, the Voatz application reports all detected anomalous events to Zimperium – not just the critical ones. These events are combined into a database that offers a view into threats seen by applications from other Zimperium customers around the world. As new threats are detected, their incidence can be tracked and, if classified as critical (e.g., Zero-Day attacks), the voter may be required to update their Voatz application to incorporate security patches.

Network Security

The guaranteed delivery of messages between the smartphone and the blockchain (see below) is critical. Voatz employs algorithms approved by the National Institute of Standards and Technology (NIST) to ensure ballot delivery. This screen is only shown when the voter's ballot has been transmitted and stored without error.

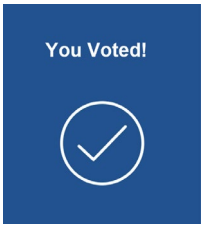


Figure 1: Proof of guaranteed ballot delivery and secure storage on the blockchain

HTTPS and End-To-End Encryption

All communication between the user's smartphone and the backend systems is encrypted using the NIST-approved algorithm Advanced Encryption Standard operating in Galois Counter Mode (AES/GCM) with a 256-bit length key. The Voatz platform uses the TLSv1.2 protocol to establish both vote capture smartphone-to-server and server-to-server communications.

Each voter's mobile device creates a public and private encryption key pair during the voter sign up process. The server also creates a unique public and private key pair to authenticate each voter. The device and the server exchange public keys during the initial handshaking process using the Ephemeral Elliptic Curve Diffie-Hellman (ECDHE)[iii] anonymous key agreement protocol[iv]. The voter's private key is securely stored in the smartphone's hardware (the secure enclave for iOS or keystore system for Android). All traffic sent over HTTPS is first encrypted by the sender using the recipient's public key in such a way that only the entities holding the private keys can decrypt one another's messages.

At the application layer, Voatz combines a SHA256 bit hash with an AES block cipher using GCM (Advanced Encryption Standard with Galois/Counter Mode) to encrypt the smartphone-to-server and server-to-server messages. At the transport layer, PKI-based (Public Key Infrastructure) payload encryption is employed for relevant API calls.

Perfect Forward Secrecy (PFS)

While encrypted traffic is unreadable, it may still get stored on certain devices, even when caching is disabled. If a private key used to encrypt traffic is compromised, that key can be used to read all previously stored messages. To prevent this kind of compromise, Voatz uses Perfect Forward Secrecy (PFS) to generate a one-time session key that is unique for each communication session. Thus, if the key for a specific session is compromised, it will not compromise data from any other session. The practice of PFS is used in many modern social media communication applications to reduce the risk of accidental data exposure.

Application Key Sequencing

Application Key Sequencing is used to maintain a logical sequence of events, starting with the mobile device or tablet activation, and all subsequent communications between the mobile/tablet application and the server. This detects situations in which both a rightful voter and an attacker are using the same account in parallel from different phones, or if the same voter is trying to register on more than one device. The key element here is the NEXT_KEY value, which is first generated randomly by the server upon activation and stored in the application's private data storage during the initial handshake. Upon each subsequent session establishment (login), the application sends this value to the server for validation. After this stage, the new value (for future sessions) is calculated by the server, then passed to the mobile application as NEXT_KEY and the device confirms that it has been received. The mobile application will use this value the next time it needs to establish the session.

Certificate Transparency

Certificate transparency is an emerging standard designed to be able to check or audit the certificates presented during the setup of an HTTPS connection. The process starts when a host/server sets up an HTTPS certificate issued previously by a trusted Certificate Authority (CA). Certificate Transparency aims to have close to real-time monitoring to find out if that certificate has been revoked or issued maliciously or by a compromised certificate authority. When a certificate is issued, the certificate authority must submit the certificate to several append-only certificate logs, which can later be cross-checked by the client and scrutinized by the owner of the domain. The certificate must exist in at least two logs for the certificate to be valid. Details about how log proofs work are described here: https://www.certificate-transparency.org/log-proofs-work.

Certificate Pinning

To prevent Man-in-the-Middle attacks, where legitimate traffic is intercepted and altered between the voter and the voting system, the Voatz application implements certificate pinning, which checks the server's certificate against a local copy of the expected certificate. The digital certificates are refreshed as needed during periodic application updates.

Input Sanitization and Validation

Voatz implements secure programming practices based on a "design by contract" model with proper data input and output validation. Thus, if the interface says it will return a "number," it should return a number and no other characters. If the server is expecting a string of less than or equal to 24 characters, the platform ensures the interface will only return up to 24 characters. This helps prevent innocent errors and, more importantly, can reduce the likelihood of various injection and memory corruption attacks.

DDoS Attack Mitigation

Voatz protects its infrastructure from distributed denial of service (DDoS) attacks, which are malicious attempts to disrupt normal network connectivity and system availability by flooding the target infrastructure with illegitimate Internet traffic. The Voatz platform uses a highly resilient 32-node cloud infrastructure, built across multiple service providers residing in availability zones in the United States, including:

Cloud Service Provider	Security references
AWS – Amazon Web Services	AWS Cloud Security Overview and security resources[v]
Azure - Microsoft	Microsoft Trust Center[vi]

Table 1: Cloud Service Providers and Security References

Continuous DDoS mitigation involves establishing a secure perimeter around the critical infrastructure and allowing or denying certain traffic based on filters or rules. The platform leverages multiple capabilities including Cloudflare to absorb and deflect unwanted traffic. Key services employed in the DDoS attack mitigation strategy include:

DNS Redundancy

One of the most common targets of DDoS attacks is the Domain Name System (DNS). Voatz uses highly available and scalable DNS service providers designed to route users to the optimal endpoints. This approach enables traffic to be managed through a variety of routing types and provides additional advanced routing capabilities to protect domain names from DNS-based DDoS attacks. Voatz also uses DNSSEC[vii] to ensure the security of its DNS table entries.

Multiple Points of Presence (PoPs) and Geo-blocking

Voatz distributes traffic across multiple PoPs and filters requests to ensure that only valid HTTPS requests are forwarded to backend hosts. This increases platform resilience and ensures legitimate traffic can reach its destination with minimal friction. The platform, optionally, also utilizes geolocation restriction, known as "geo-blocking", which is useful for isolating attacks originating from particular geographic locations.

Web Application Firewall (WAF)

Firewalls help protect web applications from common exploits that can affect application availability, compromise security, or consume excessive resources. Depending on the type of election and threat patterns, Voatz deploys customized web security rules to control which traffic can access which endpoints. Web security rules that target specific DDoS request patterns can be very effective for minimizing the effect of a DDoS attack.

Elastic Load Balancing (ELB)

Load balancing enables the automatic distribution of application traffic to several Voatz servers across multiple Availability Zones. This technique minimizes the risk of overloading the instance of a single server. Elastic Load Balancing only supports valid TCP requests, so DDoS attacks such as UDP and SYN floods are not able to reach the platform.

The table below summarizes the common threats to mobile devices detected by Voatz.

Threat Examples	Prevention / mitigation methods
Device security	Mobile Threat Defense Services
Jailbroken or "rooted" smartphone	Device vulnerability assessment is performed twice: 1. During onboarding, as a convenience, to inform the voter that their device is insecure *before* they begin ID proofing, and 2. When the voter submits their ballot to prevent ballot delivery from an insecure device.
Malicious process detection (i.e., detection of unnecessarily elevated privileges)	
Unsigned applications (i.e., detection of applications not digitally signed by Apple or Google)	
Network Security	
Man-in-the-Middle Attacks	HTTPS (AES\GCM) Application Key Sequencing, Certificate Pinning, Certificate Transparency
Data leakage	Perfect Forward Secrecy (PFS)
Multiple registrations by the same voter on different phones	Application Key Sequencing
Data injection, memory attacks	Sanitization and Validation
Distributed Denial of Service (DDoS)	Cloudflare services
DNS flood attacks	DNS Redundancy DNSSEC
HTTP flood attack	Multiple points of presence (PoPs), geo-Blocking, Web Application Firewall
UDP amplification or SYN flood attacks	Elastic load balancing

Table 2: Mobile Voting: Threats, prevention and mitigation

Secure Storage of Voted Ballot

Voatz uses blockchain technology for the secure storage of voted ballots. Blockchain technology gained its popularity as the underlying security technology to Bitcoin, a digital currency. In the past ten years, the pace of blockchain innovation[viii] has been fueled by hundreds of millions of dollars in venture capital investments[ix].

From the point of view of an attacker, the most efficient method of casting doubt or changing the outcome of an election is to attack the place where voted ballots or election results are stored. To ensure against tampering, or even a credible assertion of tampering, Voatz stores the votes on ballots submitted from smartphones on a blockchain network. Voatz selected the blockchain as its ballot storage architecture based on four criteria:

1. **Extensively vetted** by NIST[x] and major organizations like IBM, the Federal Reserve Board, and the World Economic Forum.
2. **Geographically distributed servers** – the Voatz blockchain network can be configured by the jurisdiction. Recent pilots employed 32 servers managed equally by Amazon Web Services and

Microsoft Azure cloud service providers; each provider split their 16 servers equally across two U.S.-based data centers.

3. **Redundant** – each server synchronizes with all other servers to maintain an identical copy of the votes submitted by eligible smartphone voters from virtually anywhere in the world.
4. **Immutable** – once a block of votes has been added, any attempted modifications will be detected immediately.

Unlike a "permission-less" blockchain network, like the one used by Bitcoin where anyone can add their own server, Voatz uses a *permissioned* blockchain network Hyperledger Fabric, an open source[xi] version of the blockchain originally developed by IBM and now managed by the Linux Foundation[xii]. Hyperledger Fabric is specifically engineered for permissioned blockchains.

Conceptually, a blockchain network serves a similar role as a paper ballot – a store of voter intent. However, unlike the blockchain, paper ballots are not redundant and not immutable. Paper ballots are vulnerable to human error (e.g., misplaced ballots[xiii]), malicious physical ballot manipulation (e.g., a malicious actor filling an oval to over vote a contest, and therefore deny a vote for the intended candidate, or to mark a choice in an under-voted contest) and illegal ballot harvesting[xiv] and to natural disasters like hurricanes[xv], fire[xvi] and floods[xvii].

Quick FAQs on Blockchain technology as applied to voting	
After the pilot period, will each state have its own instance of a blockchain network?	Yes. Voatz expects that counties or municipalities within the state will share the blockchain.
Who controls the blockchain?	A "certifying authority" has control over the blockchain network. Typically, that would be a state's Chief Elections Officer or their designee. So far, Voatz has been the designee in the pilots.
What does "control" of the blockchain mean?	The entity controlling a state's election blockchain can control the number of nodes (e.g., 32), the physical location of servers (e.g. only in the U.S.) and the identity of the auditors.
Is the blockchain permission-less or permissioned?	Permissioned. A permissioned blockchain network is the only practical way, for example, to vet the hosting vendor(s), to ensure that all servers are physically in the U.S., to be able to specify the consensus algorithm to ensure performance, and to enable independent auditing of the network and election.
How is consensus achieved to add blocks to the blockchain?	Voatz employs the popular "Practical Byzantine Fault Tolerant" algorithm.[xviii].
What are the contents of a block?	A block contains one or more encrypted votes but not necessarily from the same ballot (i.e. voter). Within a block, each vote is identified by a unique anonymous voter ID, the jurisdiction ID (since there can be multiple counties on the state's blockchain), the election ID (since the blockchain can, optionally, retain the history of prior elections), the contest ID, and the choice ID of the candidate receiving the vote from the anonymous voter. The contest ID and choice ID are encrypted until the polls close to prevent parties with access to the blockchain from knowing which choice is receiving votes.
How does the concept of a ledger (i.e., double-entry accounting) apply to voting?	Here is a conceptual view of how a ledger applies to voting: • The jurisdiction "creates" potential votes (analogous to unmarked ballot ovals). • When the polls open, the jurisdiction **credits** the potential votes to the eligible voter's anonymous ID. Only the choices in the contests that the voter is eligible to vote can be credited. The number of potential votes added to the voter's account is equal to the number of ovals on the voter's blank paper ballot. • When the voter selects a choice (analogous to filling in an oval), the voter's account for that choice is **debited,** and the choice's account is **credited.** • At the close of polls, the sum of credits for each choice is added to the votes for that choice made with other voting methods (e.g. precinct voting).

Table 3: The Voatz Blockchain Network: Quick FAQs

KYV™: Identity Proofing, "Liveness Test", Binding and Authentication

The development of systems to perform remote identity proofing grew out of the requirements of the Patriot Act 2001, established in response to the threats to national security posed by the 9/11 attacks. One of the provisions of the Patriot Act was the requirement for certain industries to "know your customer" (KYC). Over the past fifteen years, entrepreneurial companies have developed the capability to remotely determine if a government-issued ID is fraudulent and if the person presenting the ID is a real person. Companies, including Voatz, that provide these critical services typically follow the NIST Digital Identity Guidelines[xix] which include stringent security provisions for data exchange between service providers. Understandably, the ability to verify a remote voter's identity, called "Know Your Voter" (KYV™) is one of the more difficult challenges to mobile voting, including:

Identity Proofing

Identity proofing is a three-step process that a remote voter can typically perform in under five minutes using the high-resolution camera on their smartphone.

Credential Validation – Voatz integrates a third-party identity proofing service from Jumio that verifies the validity of government-issued photo IDs and performs a test for "liveness"[xx]. Typically, this is done one time while the credential is valid (e.g. ten years for a U.S. passport, five years for some state driver's licenses). If the voter switches to another phone due to loss or an upgrade, the identity proofing, binding and authentication steps must be repeated.

Credential validation involves taking a high-resolution photograph of a government-issued credential appropriate to the jurisdiction. In the case of a driver's license or state ID card, information on the front of the card is OCRed and compared to the information encoded in the barcode on the back of the credential. In addition, the security artifacts[xxi] found on the credential are examined for their presence[xxii].

Liveness Detection – The purpose of "liveness" test is to answer the question, "Is the person presenting a valid credential that verifies they are who they say they are, and do they exist in the real world?" This test involves asking the user to take a "video selfie" of themselves during which they must move their arms, head, blink their eyes, etc.

Photo Matching – The last step in the identity proofing process is to compare the photo on the government issued ID card against the video selfie. This match can be performed manually or automatically. If automatically, two facial recognition engines are used. If they agree that the two images are a match, the voter is verified. If not, the two images are presented for human adjudication. Once a determination is made to accept the voter's credentials or not, all personally identifying data collected during photo matching is deleted for privacy compliance.

Binding

Binding occurs during the same session as identity proofing. It is done by asking the voter to re-enter the information typically used to open their smartphone – a fingerprint, face ID or PIN. Binding ensures that only the identity-proofed voter can vote on their device and that that person cannot vote on another device in the same jurisdiction—one and only one voter per device.

Smartphone Authentication

Where identity proofing is performed episodically (i.e. not done again until the credential expires or the user gets a new phone), authentication is performed twice during the voting process – once to open the ballot and again to submit the ballot. The purpose of authentication is to answer the question, "Is the person attempting to open the ballot or attempting to submit their voted ballot the same person who presented credentials at an earlier time?"

Authentication is performed in the same way the user opens their smartphone – by an 8-digit PIN or biometrically via a fingerprint or face ID. A voter may be allowed to spoil their ballot (i.e., submit more than one ballot); authentication ensures that only the last ballot submitted by a given voter is counted.

The following table summarizes the common threats that are deterred by Identity Proofing, Binding and Authentication.

Threat Examples	Detection & prevention methods
Is the government-issued credential valid and appropriate to the jurisdiction?	The jurisdiction specifies valid credential types (e.g. driver's license, U.S. passport, etc.). The credential service provider maintains a database of credential types and verifies that the remotely presented credential, captured by a high-resolution camera, is one expected by the jurisdiction and, by examining the security artifacts on the credential, is determined to be not fraudulent.
Is the person presenting the credential alive (i.e. not a photo)?	The prospective voter is required to take a video selfie during which motion is detected. Presenting a photograph will not work.
Is the person voting the same person whose identity was proofed?	Prospective voters must authenticate themselves in the same way they did during the original identity proofing step (e.g. fingerprint or face ID).
How can we ensure that only one person can vote on only one device?	Binding a person to their device biometrically or with a PIN known only to the device's owner prevents this threat.
Is the person eligible to vote?	Only invited registered voters who have established their identity are sent a ballot. Elements of the person's identity (e.g. name, birthdate, address, etc.), are compared to the voter registration file to verify eligibility.
How is privacy preserved?	Once identity is verified, all personally identifying information related to identity proofing is deleted.

Table 4: Identity threats detected and prevented

Independent Post-Election Audits

A voter's ability to *see* how their ballot was recorded, along with a means to disagree or simply change their mind, is an important step towards building trust in the voting process. While paper ballots provide immediate feedback, they are not the best method of recording a voter's selections when circumstances make it difficult to deliver and return a paper ballot in a reliable, timely and secure manner.

The Voatz Mobile Voting System is designed for the voter to verify their own selections and, if permitted by the jurisdiction, change their mind and re-vote a ballot. It is also designed for the jurisdiction to automate the audit for every ballot submitted through the Voatz Mobile Voting Platform.

Currently, the Voatz method of post-election audit differs from cryptographic methods advocated by some academics[xxiii]. The Voatz method of voter verifiable, post-election audits features a *visual* element to the verification process. Once the usability of this method of verification is refined through pilots, a cryptographic proof will become trivial, which will help to address the failures of earlier attempts to resolve the tension between security and usability[xxiv].

Figure 1 shows the workflows that enable the voter to verify visually that their selections have been correctly recorded.

Figure 2 shows the workflow of how the jurisdiction can audit all ballots submitted through the Voatz Mobile Voting Platform that are rendered as paper ballots.

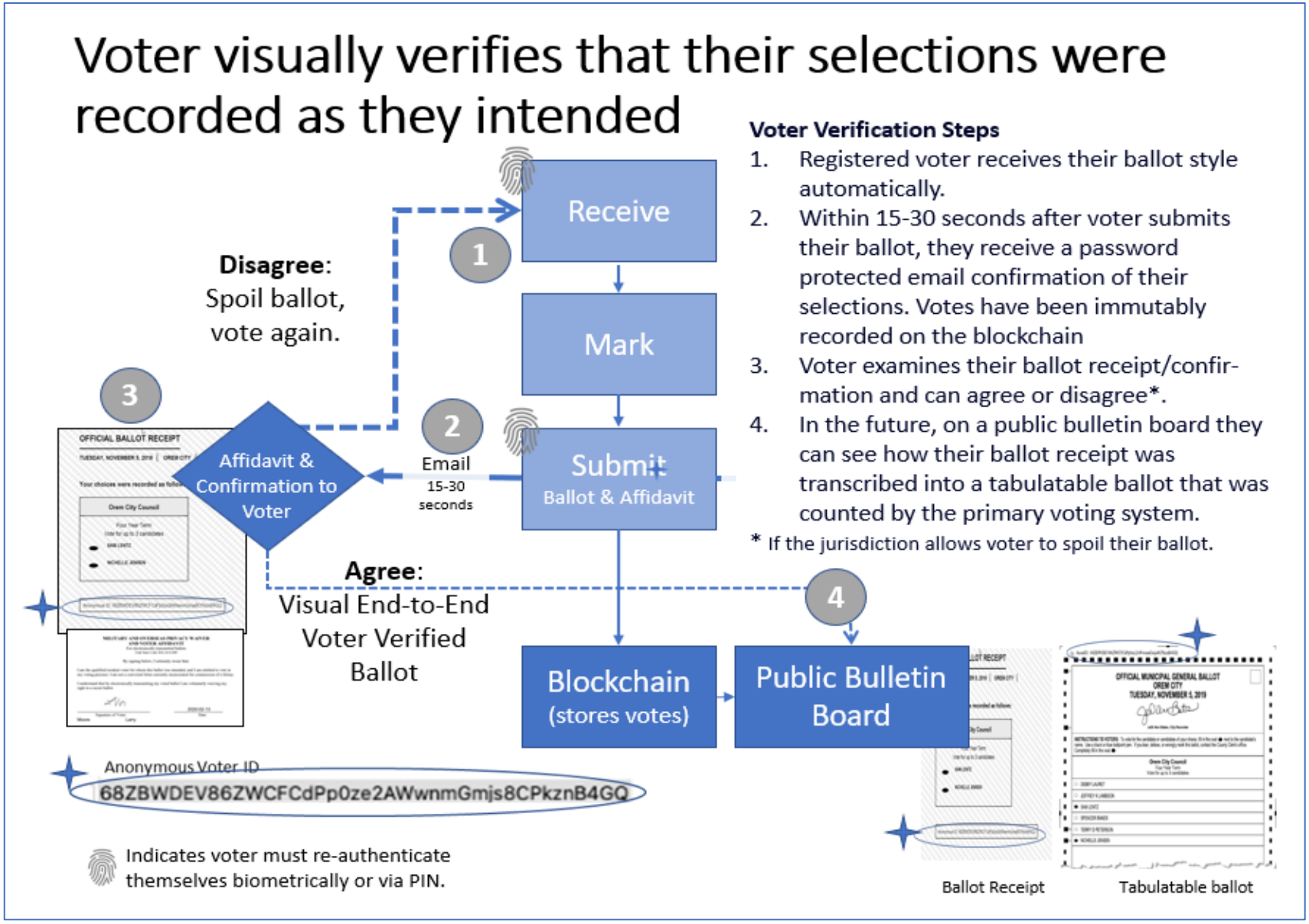


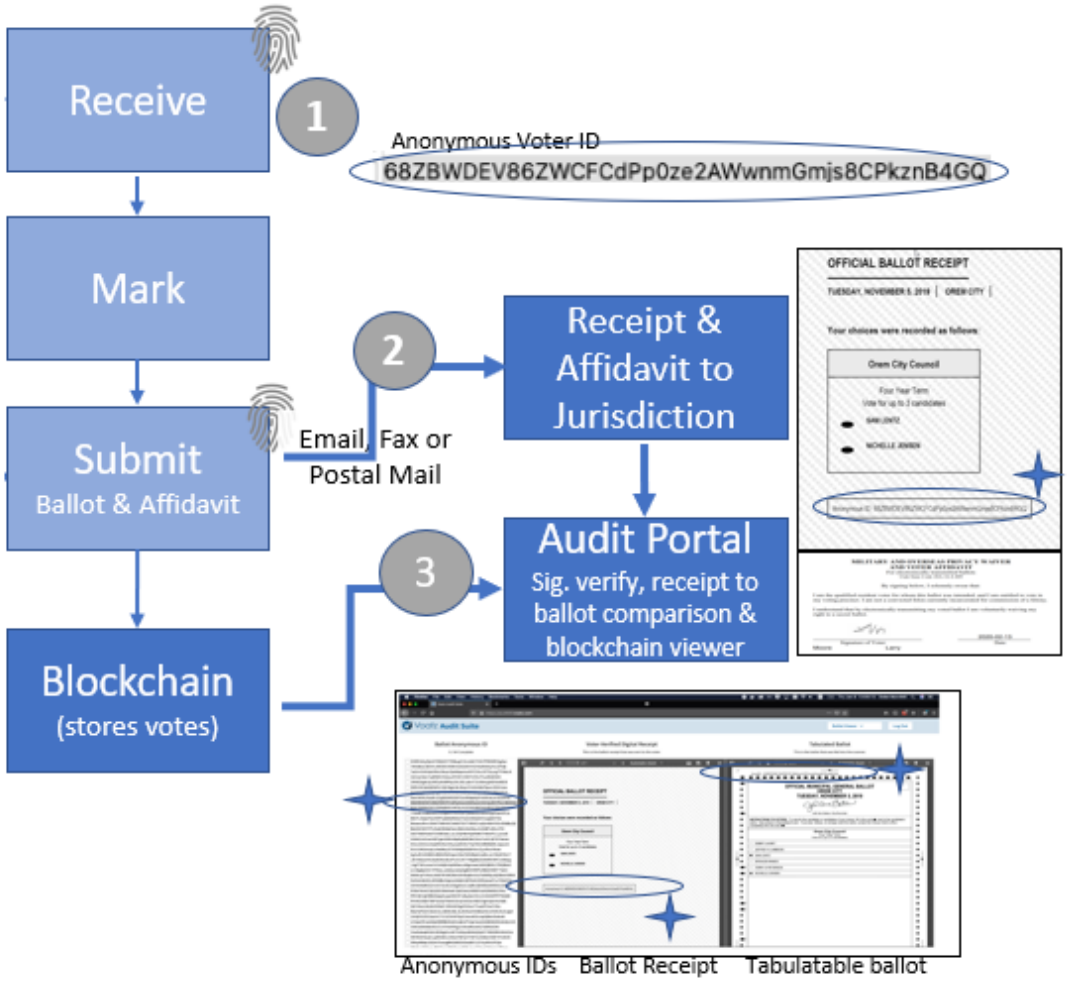
Figure 2: Voter verification that their intent was correctly recorded

Jurisdiction's Visual Post-Election Audit

Jurisdiction Post-Election Audit

1. Registered voter receives their ballot style automatically.
2. Within 15-30 seconds after the voter submits their ballot, accessibly, the jurisdiction receives a password protected email receipt/confirmation of the anonymous voter's selections. The voter's anonymous ID is contained on the receipt.
3. At the close of polls, the jurisdiction opens their audit portal and,
    a. Verifies the voter's affidavit signature against their signature on file.
    b. Prints all signature-verified ballots onto tabulatable ballot card stock. The voter's anonymous ID is printed at the top of the tabulatable ballot
    c. Runs the printed ballots through the tabulator of the primary voting system.

Post election audit: Using the Audit Portal, the auditors compare the ballot receipts to the tabulatable ballots and the tally of votes from the ballot receipts against the tally of votes for the same set of ballots by the primary voting system.



Indicates voter must re-authenticate themselves biometrically or via PIN.

Figure 3: Jurisdiction's Visual Post-Election Audit Procedures

Endnotes (all last accessed on 17 Feb. 2020)

[i] U.S. Vote Foundation & Galois, Inc, "The Future of Voting: End-to-End Verifiable Internet Voting: Specification and Feasibility Assessment". 2015 https://usvotefoundation-drupal.s3.amazonaws.com/prod/E2EVIV_full_report.pdf (See Abstract – Recommendations)

[ii] National Academy of Sciences, (see pages 101-105). "Securing the Vote: Protecting American Democracy", 2018, https://www.nap.edu/download/25120#.

[iii] Explanation of the Ephemeral Elliptic Curve Diffie-Hellman (ECDHE) can be found here: https://ldap-wiki.com/wiki/Diffie-Hellman%20Ephemeral.

[iv] A tutorial on the Diffie-Hellman Algorithm, https://www.geeksforgeeks.org/implementation-diffie-hellman-algorithm/ .

[v] Amazon Cloud Security. https://aws.amazon.com/security/ and https://aws.amazon.com/security/security-resources/.

[vi] Microsoft Azure Security Introduction. https://docs.microsoft.com/en-us/azure/security/fundamentals/overview and platform security detailed information at https://docs.microsoft.com/en-us/azure/security/fundamentals/overview.

[vii] ICAN.org, "DNSSEC – What Is It and Why Is It Important?" 3 Mar. 2019, https://www.icann.org/resources/pages/dnssec-what-is-it-why-important-2019-03-05-en. .

[viii] Gupta , Vinay. Harvard Business Review, "A Brief History of Blockchain", 28 Feb. 2017, https://hbr.org/2017/02/a-brief-history-of-blockchain.

[ix] MIT Technology Review, Orcut, Mike, April 2, 2019, "Venture capitalists are still throwing hundreds of millions at blockchains" at https://www.technologyreview.com/s/613247/venture-capitalists-are-still-throwing-hundreds-of-millions-at-blockchains/.

[x] NIST-IR (Internal Report 8202). "Blockchain Technology Overview", Oct. 2018, https://nvlpubs.nist.gov/nistpubs/ir/2018/NIST.IR.8202.pdf .

[xi] Github repository, "Hyperledger Fabric repository" at https://hyperledger.github.io/ .

[xii] Hyperledger.org, "The Hyperledger Greenhouse: Business Blockchain Frameworks & Tools Hosted by Hyperledger", https://www.hyperledger.org/. .

[xiii] Robles, Francis. "Nearly 3,000 Votes Disappeared from Florida's Recount. That's Not Supposed to Happen", New York Times, Nov. 16, 2018, https://www.nytimes.com/2018/11/16/us/voting-machines-florida.html. .

[xiv] Blinder,Alan. "Election Fraud in North Carolina Leads to New Charges for Republican Operative", New York Times, 30 July 2019, https://www.nytimes.com/2019/07/30/us/mccrae-dowless-indictment.html.

[xv] Bay County, FL, "2018 General Election – Election Day Voting", November 2018, https://www.dos.myflorida.com/media/700213/election-day-bay-polling-sites.pdf.

[xvi] Moran, Chris. "Harris County officials scramble after voting machine fire", Aug 27 2010, https://www.chron.com/news/houston-texas/article/Harris-County-officials-scramble-after-voting-1695734.php

[xvii] Nation Weather Service. "Remembering the November 1985 'Election Day' Flood", https://www.arcgis.com/apps/MapJournal/index.html?appid=0829a51e5b5d4b1787a785d8763c9156

[xviii] Baliga, Dr. Arati. "A review of blockchain consensus models." April 2017, "Understanding Blockchain Consensus Models", https://pdfs.semanticscholar.org/da8a/37b10bc1521a4d3de925d7ebc44bb606d740.pdf.

[xix] NIST Special Publication 800-63-3, "Digital Identity Guidelines" are contained in a four-volume set at https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf .

[xx] Lobovsky, Rich (Interview, SVP Business Development, FaceTec). "A discussion of the technology of "liveness detection". April 2, 2019. https://findbiometrics.com/facetec-rich-lobovsky-biometric-liveness-detection-504020/.

[xxi] Electronic Code of Federal Regulations, (CFR Section 37.15) " Physical security features for the driver's license or identification card". https://www.law.cornell.edu/cfr/text/6/37.15.

[xxii] Examples of security artifacts on Real IDs: http://www.mva.maryland.gov/secureid/SecureID-Card-Features.pdf.

[xxiii] See Endnote 1 (See Chapter 3: E2E-VIV Explained)

[xxiv] U.S. Vote Foundation & Galois, Inc, "The Future of Voting: End-to-End Verifiable Internet Voting: Usability Study". 2015 https://usvotefoundation-drupal.s3.amazonaws.com/prod/E2EVIV_usability_report.pdf