Open-Source, Asymmetric Key, Auditable Election System

Carey Briggs
Decent Elections LLC
decentelections.com
Nov 2023

Abstract. No computer-based election system has yet been described which allows individual electors the power to securely and confidently audit their own ballots, while simultaneously protecting elector privacy. As a result, all existing systems default to protecting privacy and require trust on the part of the elector. We propose a solution to the trust problem in public elections systems using cryptographic and other methods, herein termed the "OSAKA" protocol. The intended audience is presumed to have a working knowledge of asymmetric cryptography and hashing.

1. Introduction

When discussing security in the design of an election system, several notable characteristics gain focus. Traditionally, a secure election system must be:

Anonymous No person shall be able to deduce how any elector voted (each ballot is anonymous). This is largely a check against intimidation and/or suppression of the vote. That is to say, an individual or entity hostile towards a candidate or policy cannot search for electors who cast ballots for said candidate or policy (with the implied intent of retribution).

Severable No elector shall be capable of providing proof of how they voted (each ballot is severed from the elector). This is largely a check against bribery and coercion, similar to the Anonymity characteristic. Taken together, these two characteristics effectively prevent bribery and coercion from being viable attack vectors against an election's integrity.

Equal Each elector shall have exactly one counted vote according to their independent will, or zero if voluntarily abstaining. No elector is afforded larger representation than any other elector at the ballot box, unless an elector voluntarily chooses to abstain from casting their ballot. Attacks against this characteristic commonly include destroying valid ballots or adding invalid ballots.

Zero-Trust Electorate confidence shall not be contingent on faith in the authority running the election.

One functional example of a secure election system is the traditional ballot box. A chosen box with a narrow slit in its top is demonstrated to be empty in plain view of the electorate. The box is then closed, and identical slips of paper are distributed to each member of the electorate. Each participating member then secretly marks a vote on their slip of paper and folds it, concealing the ballot from view. Each elector then inserts their paper into the ballot box through the slit at the top, in full view of the electorate. Once all electors have added their paper to the box, the box is shaken to mix the votes. The box is then opened and ballots are counted in plain view of the electorate. This method satisfies all four characteristics defined above.

One substantial shortcoming of the ballot box is its scalability. The ballot box process is only viable for an electorate small enough to satisfactorily observe the entire process. Contemporary elections routinely handle tens of thousands to millions of electors. As a result, necessarily, the process is parallelized and distributed. Ballots are transported from local polling stations to tabulation centers for counting. A large portion of the ballot process is conduced out of view of the electorate and locked within the source code of black-box computational devices. In contemporary election systems, one or more of the characteristics listed above are mutually exclusive. For example,

no existing public election system is both severable and zero-trust.

2. Design Considerations

It is instructive to review ideal election system characteristics in reference to existing systems.

Anonymity This requirement can be easily satisfied by not linking personally identifiable information to a ballot. For example, no elector's name is printed on that elector's ballot. This requirement is ballot-facing. Specifically- no person with access to the ballots can select a ballot and discern which elector cast said ballot. If an elector randomly chooses a unique number and writes said number on their ballot, then retains the only copy of said number as a form of receipt- the presence of the unique number cannot, on its own, be used to identify the elector who cast that particular ballot. Paper ballot systems typically satisfy this requirement as no identifiable information is retained on the ballot. Most deployed electronic ballot systems also satisfy this requirement. However, this requirement is currently violated by mail-in and absentee ballots- which necessarily must accompany identifiable information to be accepted as valid and legal.

Severability This requirement is the elector-facing variant of anonymity. Given a pile of ballots, no elector can offer proof to a third party of which ballot they cast. The above example of a receipt would violate this requirement. Many electronic and paper ballots satisfy this requirement, since no elector retains information linked to a unique ballot. However, mail-in ballots violate this requirement by allowing ballots to be completed in the presence of third parties.

Equality Significant challenges arise with this requirement. An election is a massive event requiring the cooperation of many workers distributed across vast geographic regions. Any one of these workers can accidentally or intentionally misplace or destroy ballots. A malicious actor can inject additional invalid ballots, or replace valid ballots with invalid ones. Voter ID laws can help to combat attacks on equality by electors, but they can't completely prevent injection of illegal ballots by other malicious actors (e.g, dishonest poll workers or election software companies). Most ballots (paper and electronic) are highly susceptible to these kinds of attacks.

Zero-Trust For modern elections, it is impossible for every elector to observe the entirety of the chain of custody. When dealing with paper ballots, this leaves the vast majority of electors with no provable confidence in their ballot being accurately tallied. Further, even if a single elector observes the entire chain of custody of their individual ballot, no elector can be given reasonable evidence that equality is satisfied in general. All existing electronic systems similarly leave the elector with no reasonable evidence. A single bad actor can break equality.

3. Zero-Trust

Trust is difficult quality to evaluate due to its subjectivity. The traditional ballot box, itself a completely open process, is not *zero* trust. A talented magician could likely cheat the process. Even cryptography is not perfectly secure. Passwords can be guessed, rainbow tables computed, backdoors built, or worse- the mathematical foundations themselves could be found exploitable. Encryption is built upon what is believed to be sufficiently difficult problems. Cracking cryptography is merely *difficult enough* to inspire trust, as is evidenced by the trillion dollar cryptocurrency market capitalization.

4. Cryptography

We propose that the foundation of a secure election system can be built on asymmetric cryptographic functions. Digital signatures can provide proof of ownership with regard to strings of data which have been collated into a database. They also provide proof that said strings of data have not changed. This is perfect for digital ballots. An elector's ballot can be rendered into a predefined format (XML, etc) and then signed by said elector's unique, ephemeral private key. The ballot, public key, and signature comprise the elector's ballot package. The elector then sends their ballot package to a central collation server where it is validated and stored. At the conclusion of an electoral period, the collated database is released publicly for any interested party to compute the results of an

election. Any interested elector could verify their ballot is included in the database by simply retaining their ballot signature and private signing key at the time of voting and search for it in the database. If the associated ballot data (XML, etc) matches the signature and key, the elector has compelling evidence that their ballot hasn't changed from its casting. By retaining a token, provided it can be shown to be unique, the elector can be certain their true vote is being counted toward the final tally. Their vote cannot be altered or destroyed without detection. Unfortunately, this method breaks a different requirement: Severability.

5. Severability

While severability remains a characteristic of an ideal election, it carries with it a compromise. There is no known method of ensuring complete transparency in the election results while simultaneously leaving an elector incapable of proving to a third party how they voted. A completely transparent election process might necessarily leave electors vulnerable to coercion.

The legal sphere of voting has changed in recent years. In a 2016 ruling by the First Circuit Court of Appeals, publicizing a photograph of a filled-in ballot (a *ballot selfie*) was found to be protected speech under the First Amendment. The rise of digital cameras, mail-in voting, and social media has prompted many states to legally protect the act of an elector voluntarily disclosing the contents of their ballot. As of 2023, 24 states have legally enshrined ballot selfies into law. These statutes effectively kill the perceived necessity of a voting system to be severable. It can be argued that imposing severability does more harm than good; not only by potentially concealing large-scale fraud in an attempt to prevent a smaller crime (coercion), but also by removing a direct method of redress for electors impacted by fraud.

6. Equality

The cryptographic exchange described in section 4 is incomplete: It only provides evidence to an elector that their ballot accurately reflects their will and counts towards the election results. The elector still needs a manner of redress when their ballot is altered or deleted. Simply possessing a key and a hash does not prove the holder submitted a ballot related to said hash. If an elector cries foul and insists a ballot has been deleted from the publicly released election results, all other electors must be capable of testing the claim.

To be publicly auditable, the protocol for such a voting system must be openly disclosed. As such, any programmer could generate a ballot and signature and falsely claim the ballot signature should be in the database. However, if the collation server maintains a signing keypair, and releases the public key before an election, the server can use the keypair to sign ballot signatures which have been legitimately submitted. In doing so, a third party can test deletion allegations by checking if the retained ballot signature has itself been signed by the election authority. A ballot signature, further signed by the collation server, is herein referred to as a *ballot receipt*. It serves as proof to any party that a ballot with a specific signature has been received by the collation server and must be present in the publicly auditable database released at the conclusion of any election. This provides the necessary path towards redress for an elector that has been affected by ballot tampering. It is important to note that this method can not only detect fraud, but *correct* it.

7. Hardware and Black Boxes

A common source of voter fraud allegations with respect to voting machines stems from their implementation as “black boxes”. I,e, while the internal mechanisms and processes may be described, the machine itself cannot be inspected to verify those claims. The box is effectively “opaque” and the internals cannot be observed. This is mostly by necessity. It is not possible to provide every potential elector access to every voting machine to prove the machines are functioning as intended. Further, providing such access can create opportunity for an attacker to compromise a machine into performing other than intended. This is why open-sourcing voting machine code is a meaningless gesture. One still needs to be convinced that the code a voting machine is running is the same code that was open-sourced.

The OSAKA protocol is resistant to subversive code by providing the elector with a cryptographic receipt that can be checked after submitting a vote. However, alleging misconduct on the machine's part after a ballot has been submitted carries significant problems. First: the ballot has already been submitted. The elector should have oversight on their ballot and signatures before transmitting them to the collation server. Second: nothing prevents a subversive actor from submitting a ballot and receiving the correct receipt, then alleging falsely that the process was compromised. These false allegations can be detrimental to the public perception of accountability in a process. The best solution is to render as much control as possible to each individual elector over their ballot generation. Ideally, this requires each individual elector to maintain custody of the hardware used to generate their ballot. A viable path to achieving this goal involves leveraging personal smartphones to perform most of the OSAKA protocol. By turning a cellphone into a voting machine, the election authority surrenders a vast amount of oversight to the electors. Multiple organizations with opposing interests can develop phone applications to implement the OSAKA protocol and manage ballot receipts. Electors, should they posses the proficiency, have the opportunity to download and compile their own apps, or write new ones. Data integrity can be verified at every step of the process- especially as a final check before submitting ballots to the collation server.

Black boxes effectively evaporate when every step of the process can be directly observed by each voter. But what about the collation server? It can still return a receipt for a changed ballot and incorrect ballot key while maintaining it acted in good faith.

8. Data Attestation

Ensuring equality not only requires ensuring that valid ballots are protected, but that invalid ballots are not injected into the database (stuffing the box). Since the collation server holds the election signing keys, an attacker with access could simply generate ballots and inject them into the database, producing what appears to be valid receipts for invalid ballots. A poll worker or server administrator could search for voters who have not submitted a ballot before the close of an election and use those identities to generate and submit fraudulent ballots. A potential solution to this problem is biometrically derived encryption keys. This would tie an elector's identity to their body, preventing unauthorized use and ballot stuffing. While biometrically derived keys may be a secure method of enforcing a one-vote-per-elector system, the technology may not exist or may be undesirable to implement due to privacy concerns. As such, the protocol described herein relies on an alternative: Attestation.

To ensure a ballot was generated by its purported elector, an identity attestation is required from a poll worker. Prior to an election, each registered elector is assigned a random string of bytes known as the *identity challenge*. On registration the elector also generated an identity key- keeping the private key secret and delivering the public key to the registrar to maintain. Signing the challenge bytes with the identity key provides some evidence that the elector was involved in generating the ballot. Each poll worker assigned to perform identity checks likewise generates a poll worker key- keeping the private key secret and transmitting the public key to the election authority. On checking in to a polling location, an elector presents their photo ID to a poll worker. The poll worker then signs the elector's identity challenge with their own key, creating the identity attestation. This process creates a chain-of-custody for ballots by providing a digital "sworn statement" that a poll worker has performed a voter ID check on that particular elector. Should a post-election audit reveal invalid electors or obvious fraud (such as deceased individuals submitting a ballot), the identity attestation will provide clear evidence of which individuals are responsible for the fraud.

While the threat of punishment may serve as an effective deterrent to false attestations, the deterrent is only as strong as the chance of being discovered. To increase that chance, attestations should only be possible during election hours (to prevent mass-signing false attestations prior to the election) and be done in public view. This can be forced by assigning a random *location challenge* to each polling location. On election day, prior to the polls opening, the poll workers at a given location will collectively generate a random string of bytes. Each worker may elect to add their own random seed such that each worker has an effect on the final value. In addition, election observers may be given the opportunity to add a seed as well. Once all seeds are added, the location challenge is finalized. During identity attestation, the location challenge is added to the identity challenge before being signed. This proves that the attestation occurred after the seeds were provided and not days prior to the election.

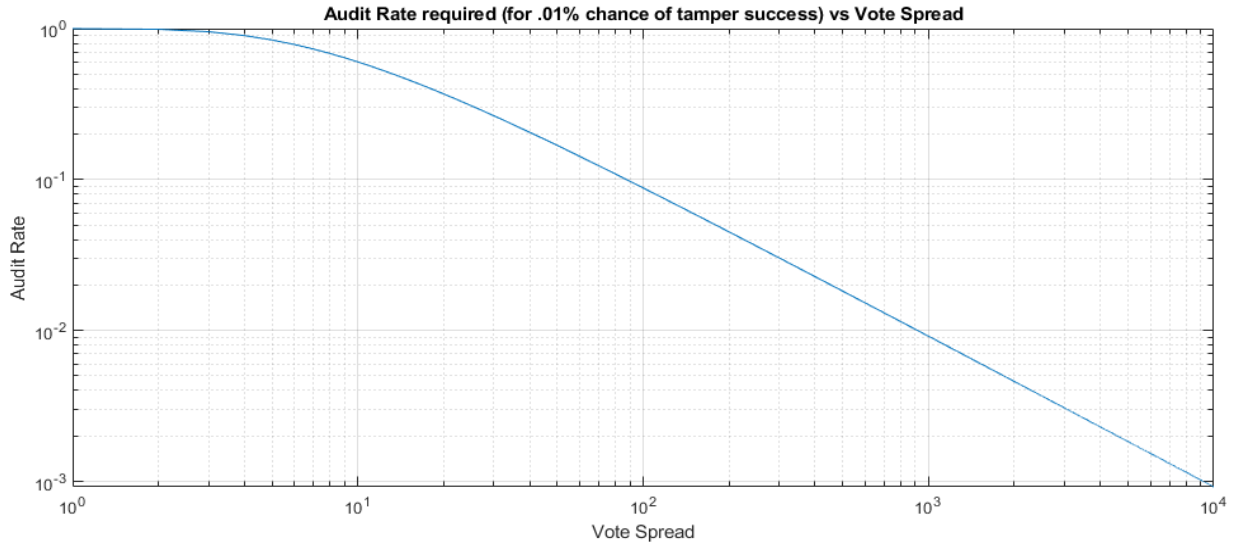
To ensure the ballots originate from a polling location (in public view), attestation machines are delivered to each polling location. These machines contain protected keys, such as those stored within a Trusted Platform Module (TPM). Prior to the start of the election, poll workers transmit the location challenge to each provided attestation machine. After an elector completes their ballot, they transmit it to a local attestation machine (such as through Near-Field Communication). The attestation machine checks that the ballot contains the location challenge, then signs it with the machine's private key- creating the *location attestation*. This attestation provides evidence that the ballot in question was possessed by a person physically at a publicly viewable polling location during the hours of the election and could not have been bulk-injected into the database by the election authority or an attacker sitting at the server.

9. Audits and Confidence Metrics

When the voter registry has been adequately audited for invalid entries, the remaining avenue for voter fraud is flipping or altering cast votes. Ballot receipts are designed to prevent this but the effectivity depends on the audit participation rate. Given a set number of votes which need to be flipped to alter a race, each successive elector who audits their vote increases the probability of detecting the fraud. If the number of electors who have successfully audited their vote is known by an entity, that entity can compute the overall probability that enough votes have been flipped to alter the outcome of the election.

At the close of an election, the ballot database is openly published by the election authority. Despite this, most electors will likely not wish to download the entire database in order to check their single ballot. More likely, auditors will be established to provide verification services for the vast majority of electors. An elector can anonymously submit their ballot receipts to one or more auditors, who then check the ballot database for the entry encoded into the receipt. The auditor performs the necessary cryptographic checks on the ballot entry in the database, then returns the entry to the elector as verification their vote was accurately counted. Auditors can coordinate their efforts and pool ballot receipts to arrive at a reasonable estimate of the total audit participation rate. This audit participation rate can then be used to estimate the probability that the results of a particular election were unaltered by vote tampering.

While the probability of tampering depends on the number of votes audited and the total number of votes, a simplified calculation can estimate the upper-bound as the number of ballots approaches infinity. The simplified estimate of an election outcome being altered is (1-A)^n where A is the audit participation rate and n is the number of votes needed to be flipped to alter the outcome. While it is impossible to know the number of votes required to be altered to change the outcome after the outcome has already been changed, it is at least as large as the spread between the winner and runner-up. This allows auditors to estimate the maximum expected probability that an election was effected by fraud using (1-A)^n where n is the spread between candidates. As the spread between candidates narrows, the audit participation rate must increase to maintain a threshold confidence level in the election outcome. However, the audit participation rate for common vote spreads between candidates is surprisingly low:



The logistics of hosting auditing services raises voter privacy concerns. In contacting an auditor, the auditor may surmise that the requestor is the voter who submitted the linked ballot, and then connect the ballot plaintext to that voter identity through the use of metadata such as the requestor's IP address. This type of privacy leak can be mitigated by using designated proxy servers. If auditors release a public key, an elector can use that key to encrypt a request package that only that elector can decrypt. The elector sends the request cipher to an auditor proxy, which then anonymously forwards the request cipher to the auditor chosen by the elector. The elector can use a return key packaged in the request cipher to encrypt the requested ballot entry, allowing it to be relayed back to the elector without the proxy inspecting the contents. This separates the elector's identity and ballot by blocking the elector's identity from the auditor and obfuscating the ballot contents while it passes through the proxy.

10. Data Structure

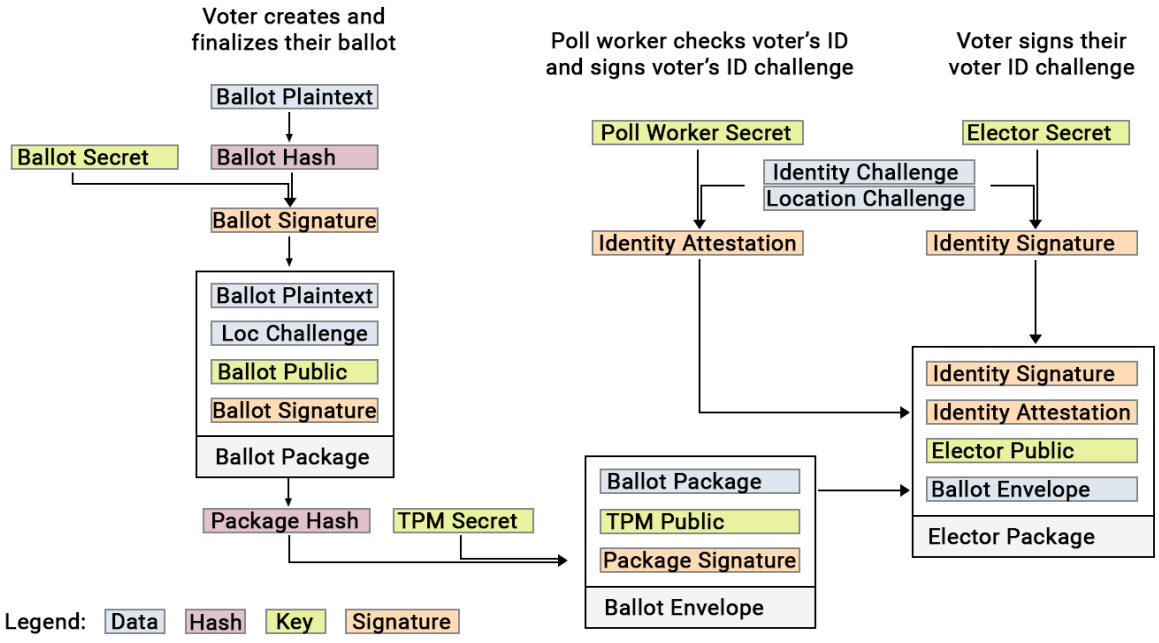
The formatting and construction of a ballot package (to be submitted to a collation server) is as follows:

Electors may download their regional-specific ballot templates prior to the election to allow sufficient time to complete it. While the ballot plaintext may contain reference IDs for candidates and measure positions, the full text of all contests and options should be documented in the plaintext as it would be presented to the elector. This limits the possibility of index swapping errors.

An elector visiting a polling station with the intent of casting a ballot first checks in with a poll worker. The elector transmits their public identity key to the worker, who then checks the elector's voter registration and retrieves their identity challenge bytes. If valid, the worker checks the elector's photo ID, establishing the elector's identity matches their identity key. The worker then combines the elector's identity challenge bytes with the polling location's challenge bytes and signs the result. This identity attestation is transmitted to the elector.

A one-time-use signing keypair is generated by the elector. The ballot plaintext is hashed and the hash is signed with this ballot private key, resulting in the *ballot hash signature*. The ballot plaintext, location challenge, ballot hash signature, and ballot's public signing key are concatenated into the *ballot package*.

The elector chooses an attestation device to sign their ballot. Optionally, the elector has an opportunity to verify the attestation machine is valid by issuing it a challenge. The elector transmits the ballot package to the attestation device, which signs it with its key, resulting in the *package signature.* The package signature is returned to the elector. The elector concatenates the ballot package, attestation device public key, and package signature to create the elector's *ballot envelope*. The elector then concatenates their identity signature, identity attestation, identity public key, and ballot envelope to create their *elector package*. The data structure is illustrated below.



The elector package is obviously encrypted for transmission to the collation server to maintain ballot secrecy.

13. Conclusion

We have proposed a method of designing an election system that is anonymous, auditable, and requires minimal trust on the part of the electorate. All election data is publicly available for any interested party to inspect. Asymmetric cryptography provides proof of tampering and evidence of ballot deletion. Further, when leveraging voter ID attestation, the system is capable of enforcing vote equality and positive identification of electors. Tabulating results has been reduced to a simple set of cryptographic and SQL operations, repeatable by any interested party. Election results are deliverable nearly instantly after polls close, putting an end to the tradition of lengthening election day to election week or month. And most importantly, the proposed method is capable of restoring public faith in the electoral process: by not requiring faith at all.