

Online Voting System Discussion

Doug Blewett

doug.blewett@gmail.com

Original work Copyright © 2021 by Doug Blewett

The “mail in voting” of the 2020 election produced a record turnout. In many states ballots were sent out to all registered voters. Mail in ballots have become one of the new voting standards. Signatures were required on most of the mail in ballots. To register to vote one provides five pieces of data: name, address, date of birth, social security number, and signature samples. I always want to preserve the traditional voting systems of registration and voting/polling. If someone wants to vote at a polling location rather than by some other means, I think the polling option should always be available. At the polling locations, a signature is required to be compared with the signature provided at registration and/or the last vote. This comparison is done by the people running the polling location.

<https://www.vote.org/voter-id-laws/>

https://ballotpedia.org/Voter_identification_laws_by_state

There were news reports of irregularities in the processing of the signatures of the mail in ballots in the 2020 election. Some reports state that signature verification was disabled or the verification was done in high percentages through human intervention – the term is adjudicated. Adjudication is fairly common for many other reasons with paper ballots. My signature varies greatly day to day. Algorithms for comparing signatures are heuristics at best. Again, at polling places signatures are all adjudicated – that is examined by poll workers.


Online voting uses the tools of the Internet to allow voting from homes, residences, and anywhere one might want to cast a vote. The HTTPS protocol is in wide use on the Internet and guards against “man in the middle” attacks. Online voting can be easily implemented with what has been traditionally collected for voter registration and it will be more secure than the mail in voting systems that were used in 2020. I want to emphasize that point. Signature verification can and should be replaced by comparison of data already collected which can be exactly or precisely matched. Date of birth and the last four digits of one’s social security number are good examples. The ballot number can be used as an added factor. Where available, one could also add a code sent to the voter’s cellphone as is done by Amazon and financial institutions. Online voting and electronic storage of votes would increase turnout and provide a much quicker and trusted vote count/tabulation. A quick and transparent tabulation is important for restoring confidence in the voting system.


The chaos of the Gore v. Bush election was related to the problems one finds in counting ballots cast on physical media; usually some paper product. Debate over physical ballot media also occurred in the 2020 election. The notion with online voting is that completed ballots would be sent by the voter directly to the government hosted ballot database, all with no human intervention, no rejected or crumpled ballots, no ballots lost in the mail, and no hanging chad as occurred in Gore v. Bush. On a small computer one can easily count a million votes per second in a computer based ballot database – all without any human intervention. The vote counting software can also be published for complete transparency. It is very simple code, just a few lines that can be quickly reviewed by even novices. With online voting, one can also allow voters to review/verify votes previously cast – we discuss that later.

Consider this implementation of online voting. Ballots would be sent out as in 2020. The ballots would include a unique, human readable, “ballot number”. This ballot number would be the only change to the ballot that was sent to all voters. Numbering the ballots means that only legal ballots can be cast. The ballot number could also be emailed or texted to the voter as we want the voting process to be as green as possible – no paper. That ballot number would be included by the voter when casting the vote online. The systems would allow for only one vote per numbered ballot. The ballot numbers are never associated with the complete, “verifiable”, identity of the voter – thus keeping the identity of the voter secret in ballot casting. This uses the same technology as that used in online banking or shopping – that is login and verify.

Some of the mail in ballots used in the 2020 election did include identification codes, but there were reports of those codes were not used. The 2020 codes were not intended for human

readability and use. The ballot numbers would be assigned by each voting area and held encrypted in a voting verification database. The database would include a one way encryption (i.e. hash code) of the ballot number, a one way encryption of the registration data, a time stamp, and a flag marking if the ballot has been used. The local voter registrar is the only entity in the voting process that would have the voter identification. In the verification database the identification is done via the encryption hash. The ballot number is also never visible in the system as it is stored as a one way hash. Access to the database does not expose the voter or the ballot number.






Login for Tech Square 2021 Voting

first name: [Help](#)

last name: [Help](#)

zip code: [Help](#)

social security: [Help](#)

birthday:  [Help](#)

[Help](#)

The ballot number acts as an additional factor to verify the voter to the system. The ballot number guarantees that the voter is using a legal ballot. The system also enforces that the ballot can only be used with the assigned voter. This is more secure than the long trusted absentee balloting system we have used for decades. And those ballots are handled by many people on the way to being counted. Adding the ballot number is the equivalent of two factor verification that is, again, used with Amazon purchases or is commonly used in financial transactions.

Online voting would be a two step process: voter login and ballot casting. Voters would login to the voting system using their name, address, and date of birth and possibly the last four digits of your social security number. We would like to see voters get a registration number which could be used in lieu of a signature. Voters are limited to one login per vote – that is only one vote per voter. The login would give access to the ballot casting screen. After login the ballot screen would be displayed.


The image shows a web interface for the 'Tech Square 2021 Presidential Election'. At the top center is the Great Seal of the United States. Below the seal, the title 'Tech Square 2021 Presidential Election' is displayed in bold. Underneath the title, there is a label 'ballot number:' followed by a text input field containing the number '1026'. To the right of the input field is a blue 'Help' link. Below the input field, there are two radio button options for the presidential candidates: 'Thomas E. Dewey' (with an unselected radio button) and 'Harry S. Truman' (with a selected radio button). Each candidate name has a blue 'Help' link to its right. At the bottom center is a button labeled 'Submit the Vote', and to its right is another blue 'Help' link.

On the ballot screen the voter would enter the “ballot number”. The voter identification information and the ballot number are never included together. A doubly encrypted voter, one way, hash is sent out with the ballot screen and the ballot number is entered into that screen by the voter. The voter hash maintained by the system is never visible outside of the database. Again, the database version of the voter identity hash is encrypted (doubly encrypted ID) prior to it being sent out with the ballot webpage. Ballot numbers are checked to avoid multiple votes on a single ballot. Ballot numbers can only be used once. Voters can only vote once.

The numbers produced by the encryption hash are computationally very difficult (impossible given computer time) to link back to the identity of the voter. Think of adding all of the digits in your name, address, date of birth, and last digits of your social security number and then multiplying it by a hidden key. The hash is calculated again in real time by the login software and used by the election board when sending out the ballot collection screen to the voter. Again, the key to that encryption would change over time to keep the voter’s registration hash from being reused. All of this is standard cryptographic processing and not seen by the voter.

Completed ballots would be sent directly to the government polling website. Ballots would be entered into a cast ballot database. The cast ballots would include the encrypted ballot number. All of the processing of the cast ballots would be automated without human intervention thus avoiding the problems with physical voting media. As mentioned above, the HTTPS Internet protocol protects against “man in the middle” attacks. That is HTTPS is a secure transport that guarantees secure communication with web servers. There are also secure techniques for hiding one's IP address (e.g. Tor). There is not a pressing need to hide the IP address. People can be observed entering polling places.

The ballot number that is sent to the voter could be an encrypted number. Encryption using the advanced encryption standard (AES) or other standard would produce a small number that could be easily entered by the voter. The voter need not go through any tedious encryption processing. The voter only sees the ballot number and that number is opaque to the voter – that is has no meaning other than being a number.



Tech Square 2021 Presidential Election

Your cast ballot number is:

7de154e78efef4bbbd6d85e84da988ca7ab14e850608b4af6ba057cb

You can verify your vote using this number.

Information processing has made great strides. As votes pour in they would be “journal-ed” to avoid loss and to enable processing that we often refer to as a “chain of custody” verification and auditing. Journaling also allows for accurate vote count monitoring by region. Again, this would be all automated without any human intervention. The code for an online voting system should all be made public. There is no reason to keep the code hidden/secret.

After casting ballots, voters would like to be able to review the cast ballots as a way to personally verify that their ballot was recorded correctly. When a



Vote Verification Tech Square 2021

vote verify number: [Help](#)

ballot number: [Help](#)

[Help](#)

legal ballot is cast online, the system could return a “cast number” that could be used to later inspect the cast ballot. To review a cast ballot, the voter would use the cast number to view the cast vote. The cast ballot database would store the encrypted versions of the of the cast number and possibly the ballot number. Those two numbers could be used at a later date to retrieve and examine the cast ballot. The cast ballot number alone would provide enough information to retrieve the cast ballot. The ballot number or a password could also be added. The voter’s identity would not be revealed using this method. Legal access to the voting system would be preserved.

One has to compare online voting with mail in voting as the test for viability. There is a “consensus” argument that online

voting will never be as secure as paper media based voting. The scientific rationale for that argument is usually completely missing. Every time voting media is physically handled there is the possibility for error or worse yet fraud. From Ballotpedia, a non partisan organization, rejected ballot percentages are around one percent of the mail in/absentee ballots. The numbers for the 2020 election are still out in many locales. Consider the

many human steps involved in mailing back a ballot and that ballot being tabulated by running the ballots through tabulating machines. We need to move to online voting. It is time.

https://ballotpedia.org/Election_results,_2020:_Analysis_of_rejected_ballots



A More Detailed Explanation

The ballot number database has a hash for the voter data and a hash of the ballot number. The voter is sent and enters the ballot number in the clear (not encrypted). The voter is identified by name, address, city, zipcode, social security or driver's license ID, and birthday. All of this data is currently collected as part of the voter registration. The voter and the ballot number are represented internally with a one way hash codes most likely one of the sha-3 hashes. We are using php versions of those routines in our prototype.

<https://en.wikipedia.org/wiki/SHA-3>

As in the 2020 election, ballots are sent to all registered voters. Those ballots include the human readable ballot number. It could also be emailed or texted in the clear - no encryption required. The encryption is all done on the "election commission" server side.

When the voter decides to vote, they login to a voting web server. The server takes only HTTPS protocol requests - which avoids "man in the middle" style interventions. The voter logs in using name, city, zip, last four digits of their social security number or their driver's license number and birthday. As mentioned above, this data and these numbers are part of every voter registration in every state in the United States.

The voter's login is verified by looking up a hash of the voter's input information in a voter validation ballot database (VVDB). The voter validation ballot database contains four entries: voter hash code, ballot hash code, a timestamp for the last activity and a flag denoting if the voter has voted in this election or the ballot is currently "in use" - the voter is voting. The "in use" marking keeps the ballot from being reused via various hacking techniques.

The VVDB is produced by the registrar for the election commission for each election. This is a similar process as that which was performed by the voting officials to create the mailing list for ballots in 2020. We have a php script of less than 20 lines that creates the VVDB and the mailing list used to send out ballots - again as was done in 2020. At no time is the voter's identity exposed in the VVDB or is access enabled via the mailed ballot and ballot number.

After the voter logs in, the voter is sent a webpage with the ballot to be filled out for the current election. The ballot number is entered by the voter into the webpage and returned as part of casting the ballot. The returned ballot number is hashed by the web server for use as an index to search for the hashed ballot number and hashed voter data stored in the ballot database.

On the voting screen a hash of the hash of the voter data is included with the web page. What we are saying is that the voter identity data is doubly hashed. This double hash allows the system to verify that the ballot number and the voter ID match what was sent out. The double hash keeps the first hash of the voter data hidden. This protects against hackers repeated entry attacks. This also preserves the voter's identity and maintains the secrecy of the vote.

The hash of the voter data is located by hashing the ballot number returned in the clear. That new ballot hash is used to find the stored hash of the ballot number and the stored hash of the voter data stored in the ballot database.

When the ballot matches the hashed ballot number, the stored hash of the voter data is hashed to check for a match with the returned doubly hashed voter data. This avoids miscreants snooping the hashes and acts as an added security factor.

Prototype Implementation Details

This prototype was written to be easily installed and examined by those with an interest in the topic of online voting. This code runs without modification on Ubuntu Linux, Windows 10, and Mac OS.

This system uses flat text files throughout to simulate the databases. This was done to allow the prototype to be easily setup and examined without having system wide access to the computer resources. A “real” implementation should use a database system that allows locking on a record by record basis. This implementation uses php file locking and entries or flags in the files to lock individual records.

When a voter logs in, their record in voter validation ballot database (VVDB) is locked first by the php file lock and then by a flag added to the record that signifies that the record is in use. After the flag is added to the record, the file is unlocked. Only one voter is allowed to be logged in on each record. When the voter casts a ballot, a flag is added to the record signifying that the voter has voted. In both of those cases further use of that record and subsequent logins are disabled. Record locking is a simple, but effective technique for limiting access.

https://en.wikipedia.org/wiki/Record_locking

We run a local user level php server for debugging and testing. A php server can be invoked in the project directory with the following command:

[`php -S 127.0.0.1:8181`](#)

That will set up a web server on port 8181 - <http://localhost:8181>. Of course, if one has a running HTTP or HTTPS server running, one can place the files in a directory for that server. Again, a real system would use an HTTPS enabled server.

The prototype voter validation ballot database (VVDB) is generated using the following command:

[`php vote_generate_ballots.php`](#)

Or by entering the following in a web browser window:

http://localhost:8181/vote_generate_ballots.php

The registration data is kept in the file, "data/voter_registration.txt". This is a CSV style file with semicolon separated fields. As would be the case in an official voter registrar database, the ballot numbers are never stored in this file. The ballot numbers are generated at the time the blank ballots are created. The ballot numbers would never be recorded. The generation of the VVDB,

in a real scenario, might be handled by a separate trusted source. Again, the VVDB can be used by voting commissions and does not expose the voter identification or ballot numbering.