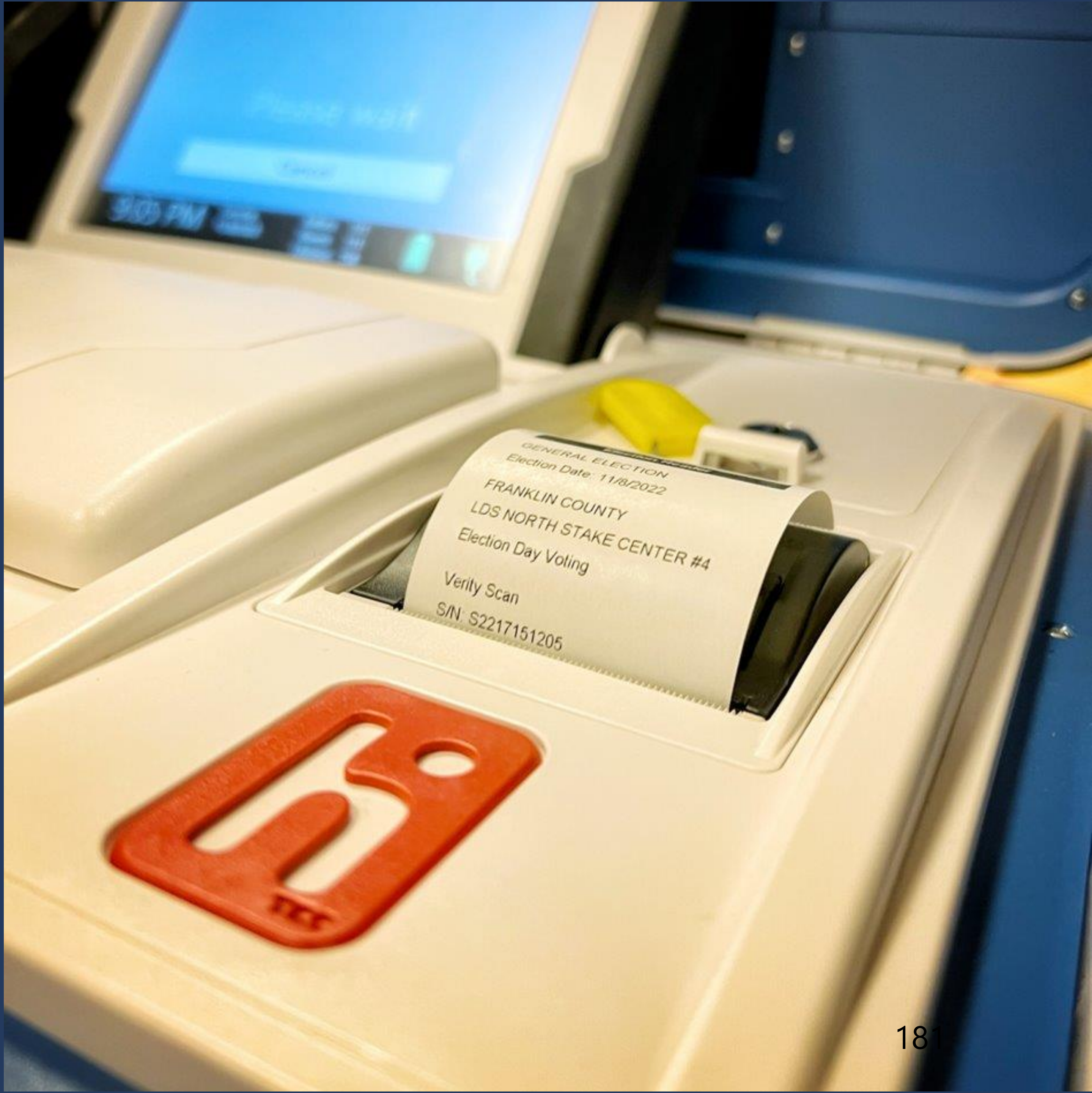
Microsoft

ElectionGuard and the College Park 2023 November Election

RC Carter, Microsoft

July 11, 2023



Agenda

- ElectionGuard overview
- How it will work this November

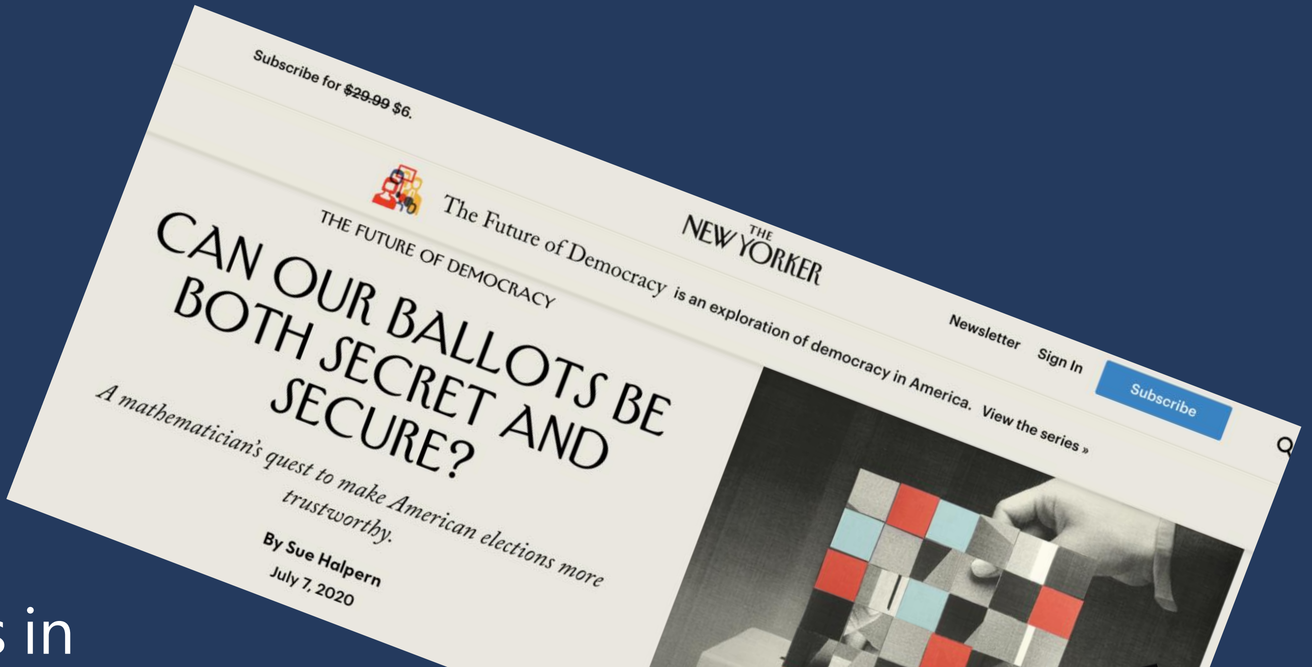
What is ElectionGuard?

- Free, open source software for voting system vendors (or election administrators) to implement *end-to-end verifiability (e2ev)*
- Runs alongside (and does not modify) the core election system
- Increases confidence in election outcomes through transparency and advanced cryptography and security
- Enables a public copy of the election results that preserves voter privacy and the secret ballot while allowing independent verification of the ballots and tallies

ElectionGuard

ElectionGuard History

- Started in 2019
- First code release September 2019
- Bounty program started in 2020
- First public election with VotingWorks in Fulton County, Wisconsin, February 2020
- Partnership with Hart July 2021
- First public election with Hart, Enhanced Voting, Center for Civic Design, and MITRE in Franklin County, Idaho November 2022


Subscribe for ~~$29.99~~ $6.
The New Yorker
The Future of Democracy is an exploration of democracy in America. View the series »
Newsletter
Sign In
Subscribe

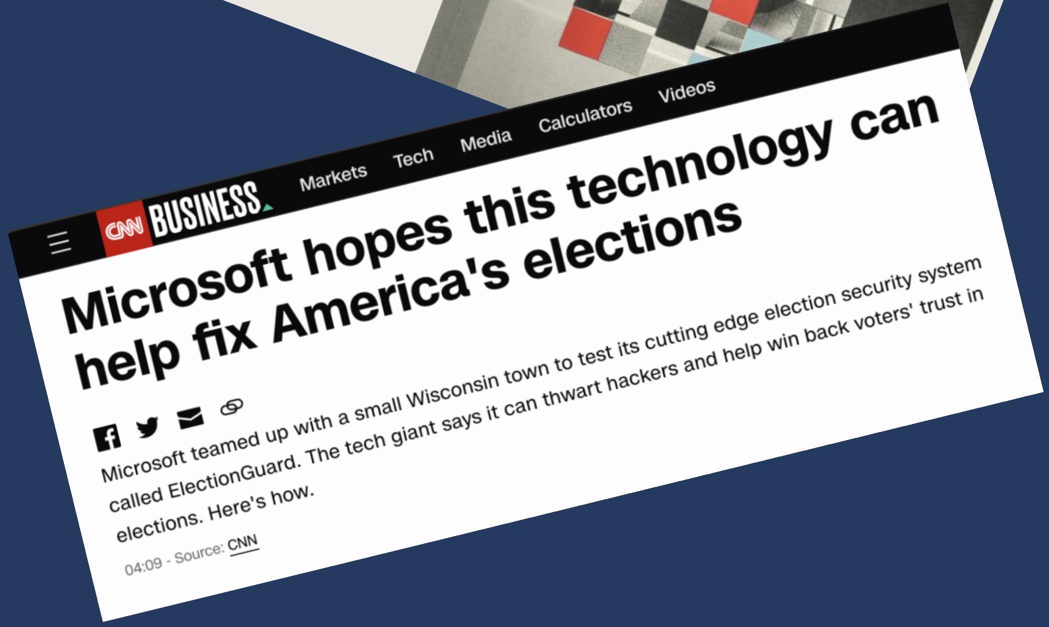
THE FUTURE OF DEMOCRACY

CAN OUR BALLOTS BE BOTH SECRET AND SECURE?

A mathematician's quest to make American elections more trustworthy.

By Sue Halpern

July 7, 2020



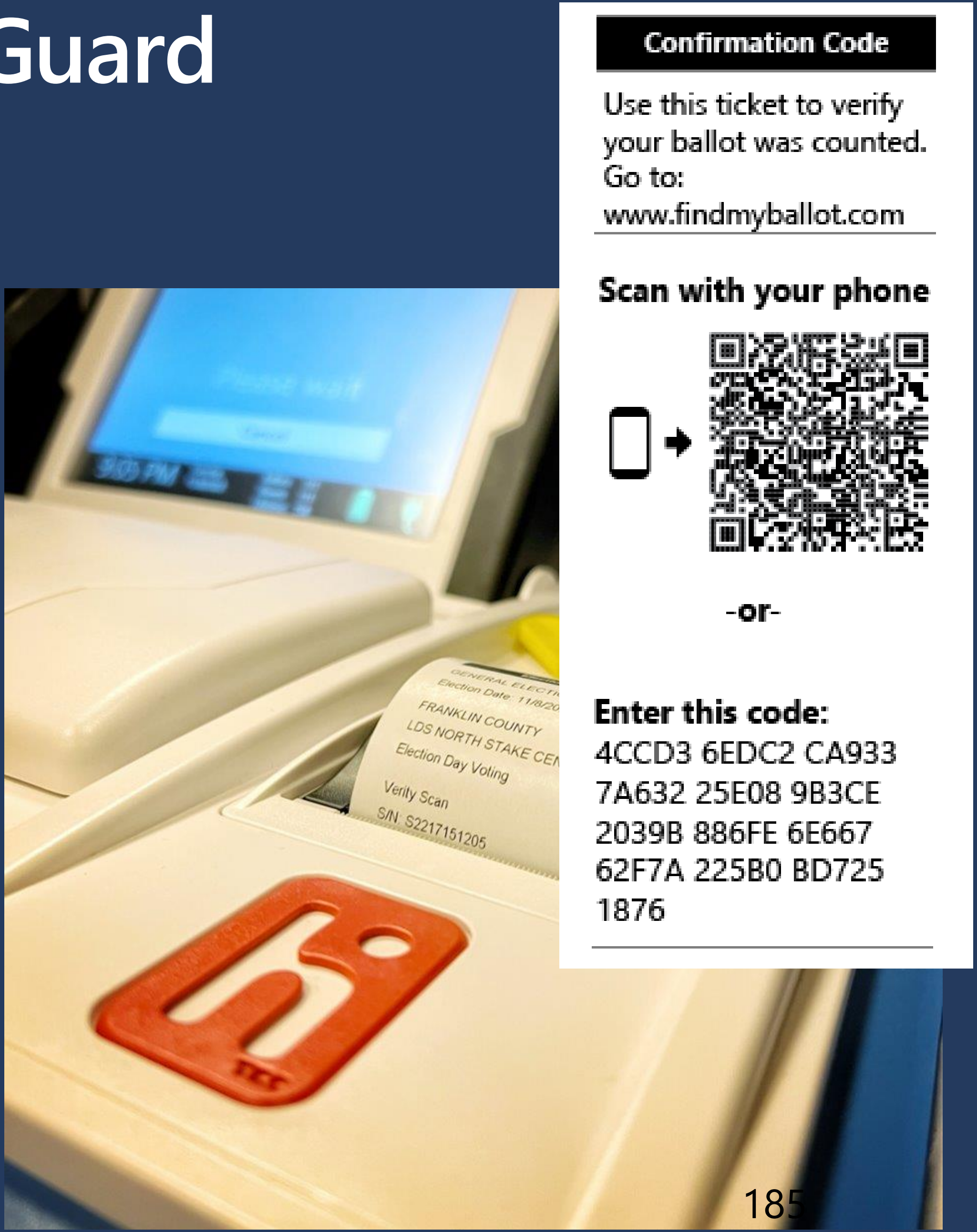
Microsoft hopes this technology can help fix America's elections

Microsoft teamed up with a small Wisconsin town to test its cutting edge election security system called ElectionGuard. The tech giant says it can thwart hackers and help win back voters' trust in elections. Here's how.

04:09 - Source: CNN

The Voting Process with ElectionGuard

- The voter votes as usual (fills out a paper ballot)
- The voter inserts the ballot into the Hart scanner
- The scanner prints the ElectionGuard confirmation code and shows a summary screen to the voter with the interpretation of the ballot
- If the voter submits the ballot, they take the confirmation code home to check that ballot was included in the published tally
- If the voter runs a BallotCheck instead, the ballot will not be included in the tally
  - BallotCheck ballots are published separately so the voter can validate what the machine *would have* submitted if the voter chose to cast



Confirmation Code

Use this ticket to verify your ballot was counted. Go to: www.findmyballot.com

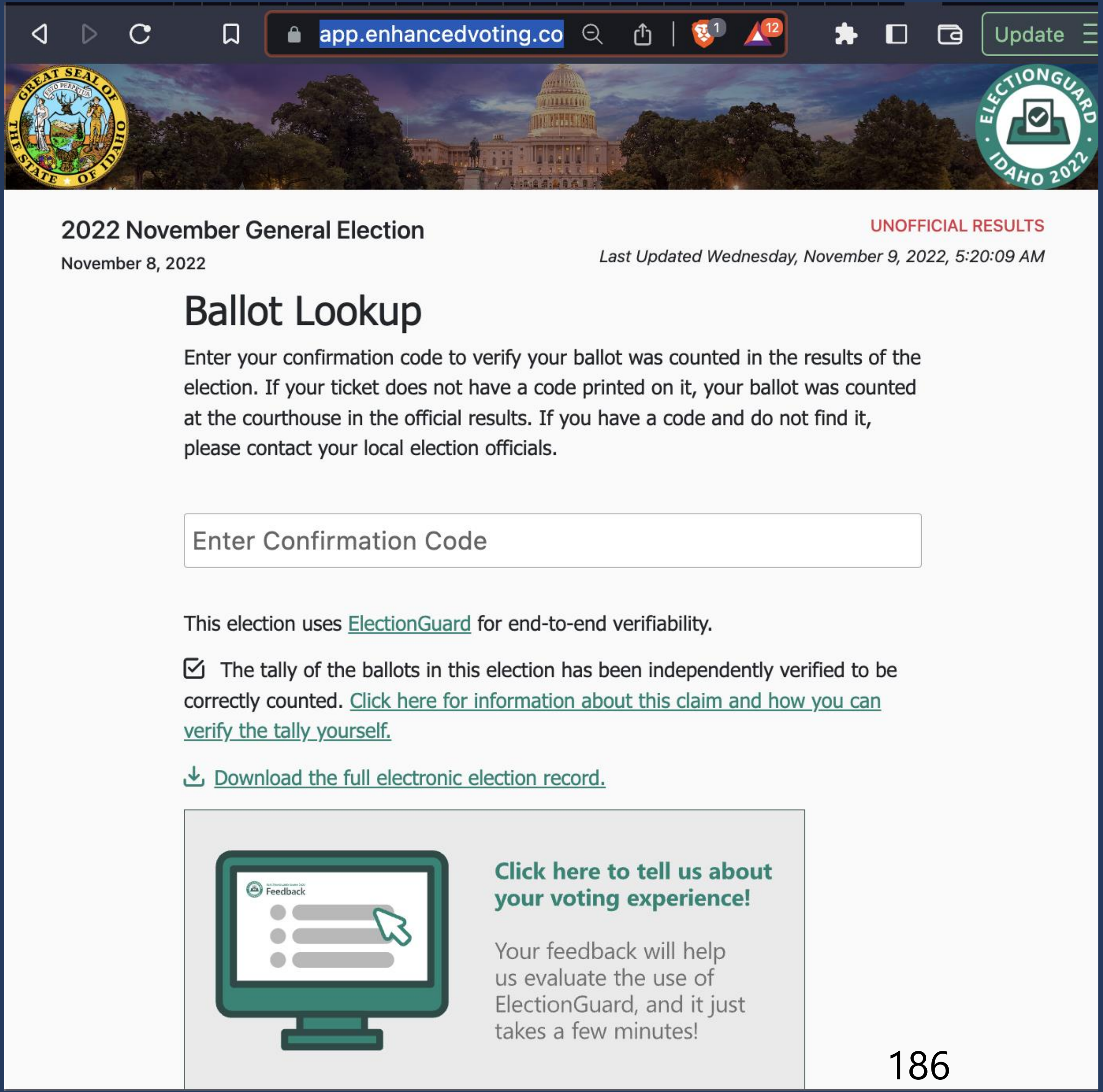
Scan with your phone

-or-

Enter this code:
4CCD3 6EDC2 CA933
7A632 25E08 9B3CE
2039B 886FE 6E667
62F7A 225B0 BD725
1876

Verifying Your Ballot was Counted

- Get the URL to the published Election Record
  https://app.enhancedvoting.com/results/public/cc/id/22
- Go to the MITRE verifier page
  https://mitre.github.io/ElectionGuardVerifier.jl/gitpod.html
- Run the verifier on a remote server (using Github Copilot)



2022 November General Election

November 8, 2022

UNOFFICIAL RESULTS

Last Updated Wednesday, November 9, 2022, 5:20:09 AM

Ballot Lookup

Enter your confirmation code to verify your ballot was counted in the results of the election. If your ticket does not have a code printed on it, your ballot was counted at the courthouse in the official results. If you have a code and do not find it, please contact your local election officials.

Enter Confirmation Code

This election uses ElectionGuard for end-to-end verifiability.

☑ The tally of the ballots in this election has been independently verified to be correctly counted. Click here for information about this claim and how you can verify the tally yourself.

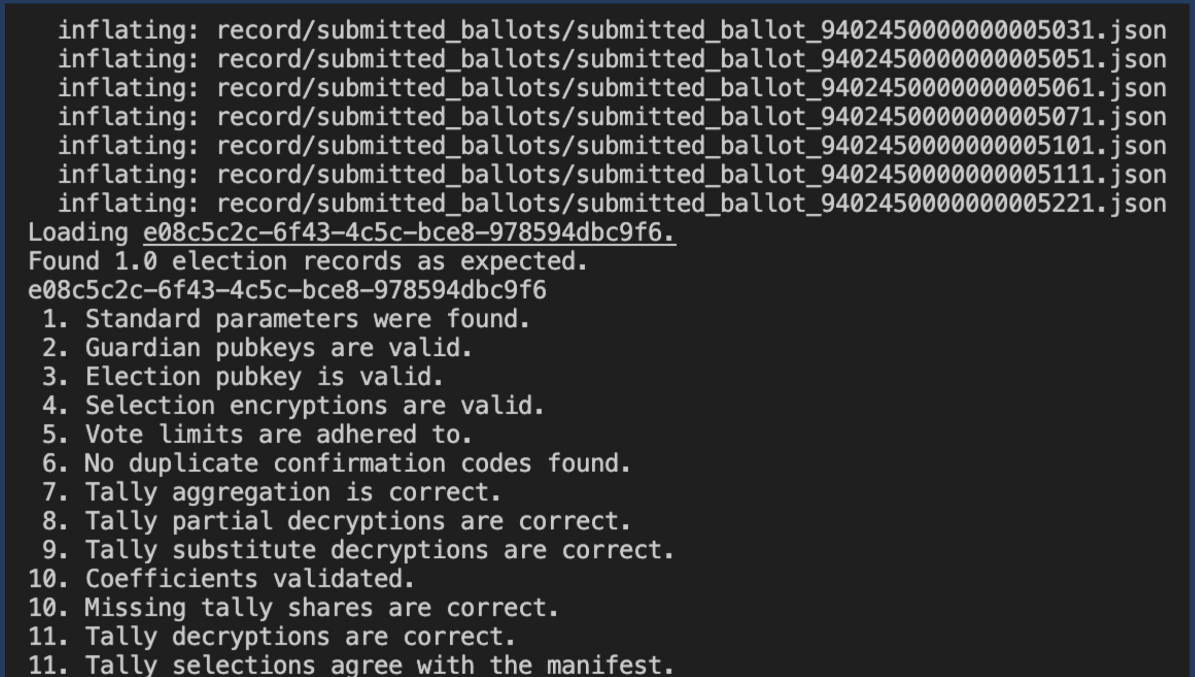
Download the full electronic election record.

Click here to tell us about your voting experience!

Your feedback will help us evaluate the use of ElectionGuard, and it just takes a few minutes!

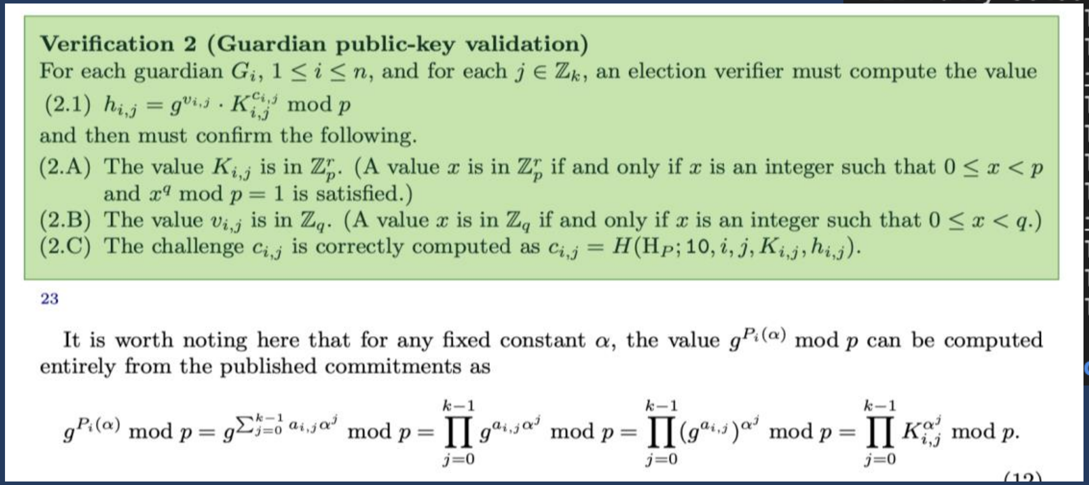
Independently Verifying the College Park Results

- Access the published Election Record from the confirmation code website
- Run an independent verifier on the published election
  - Verify each and every ballot is valid and hasn't been modified
  - Verify the tallies derived from the ballot are valid



```
  inflating: record/submitted_ballots/submitted_ballot_9402450000000005031.json
  inflating: record/submitted_ballots/submitted_ballot_9402450000000005051.json
  inflating: record/submitted_ballots/submitted_ballot_9402450000000005061.json
  inflating: record/submitted_ballots/submitted_ballot_9402450000000005071.json
  inflating: record/submitted_ballots/submitted_ballot_9402450000000005101.json
  inflating: record/submitted_ballots/submitted_ballot_9402450000000005111.json
  inflating: record/submitted_ballots/submitted_ballot_9402450000000005221.json
Loading e08c5c2c-6f43-4c5c-bce8-978594dbc9f6.
Found 1.0 election records as expected.
e08c5c2c-6f43-4c5c-bce8-978594dbc9f6
 1. Standard parameters were found.
 2. Guardian pubkeys are valid.
 3. Election pubkey is valid.
 4. Selection encryptions are valid.
 5. Vote limits are adhered to.
 6. No duplicate confirmation codes found.
 7. Tally aggregation is correct.
 8. Tally partial decryptions are correct.
 9. Tally substitute decryptions are correct.
10. Coefficients validated.
10. Missing tally shares are correct.
11. Tally decryptions are correct.
11. Tally selections agree with the manifest.
               lot 9402450000000001531 partial decryptions are correct.
               lot 9402450000000001531 substitute decryptions are correct.
               iled ballot 9402450000000001531 shares are correct.
               lot 9402450000000001531 decryptions are correct.
               lot 9402450000000001531 selections agree with the manifest.
               lot 9402450000000003191 partial decryptions are correct.
               lot 9402450000000003191 substitute decryptions are correct.
               iled ballot 9402450000000003191 shares are correct.
               lot 9402450000000003191 decryptions are correct.
               lot 9402450000000003191 selections agree with the manifest.
               lots are well-formed.

               ce/ElectionGuardVerifier.jl (master) $ 
```



Verification 2 (Guardian public-key validation)

For each guardian G_i, $1 \le i \le n$, and for each $j \in \mathbb{Z}_k$, an election verifier must compute the value

(2.1) $h_{i,j} = g^{v_{i,j}} \cdot K_{i,j}^{c_{i,j}} \bmod p$

and then must confirm the following.

(2.A) The value $K_{i,j}$ is in $\mathbb{Z}_p^r$. (A value x is in $\mathbb{Z}_p^r$ if and only if x is an integer such that $0 \le x < p$ and $x^q \bmod p = 1$ is satisfied.)

(2.B) The value $v_{i,j}$ is in $\mathbb{Z}_q$. (A value x is in $\mathbb{Z}_q$ if and only if x is an integer such that $0 \le x < q$.)

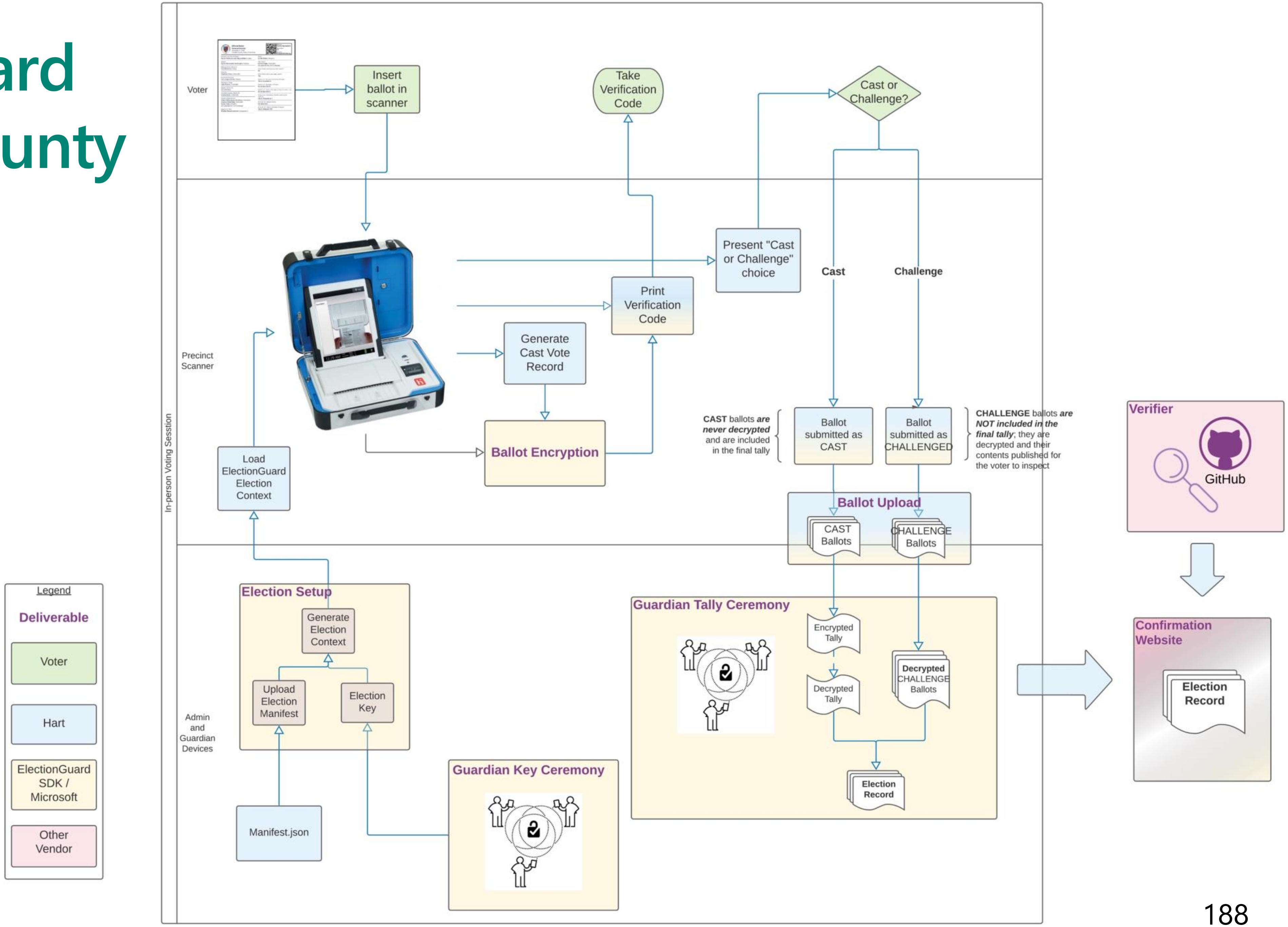
(2.C) The challenge $c_{i,j}$ is correctly computed as $c_{i,j} = H(\mathrm{H}_P; 10, i, j, K_{i,j}, h_{i,j})$.

23

It is worth noting here that for any fixed constant α, the value $g^{P_i(\alpha)} \bmod p$ can be computed entirely from the published commitments as

$$g^{P_i(\alpha)} \bmod p = g^{\sum_{j=0}^{k-1} a_{i,j}\alpha^j} \bmod p = \prod_{j=0}^{k-1} g^{a_{i,j}\alpha^j} \bmod p = \prod_{j=0}^{k-1} (g^{a_{i,j}})^{\alpha^j} \bmod p = \prod_{j=0}^{k-1} K_{i,j}^{\alpha^j} \bmod p. \quad (12)$$

ElectionGuard Franklin County Process Diagram



Legend

Deliverable

Voter

Hart

ElectionGuard SDK / Microsoft

Other Vendor

In-person Voting Sesstion

Voter

Insert ballot in scanner

Take Verification Code

Cast or Challenge?

Precinct Scanner

Present "Cast or Challenge" choice

Print Verification Code

Generate Cast Vote Record

Ballot Encryption

Load ElectionGuard Election Context

Cast

Challenge

CAST ballots are never decrypted and are included in the final tally

Ballot submitted as CAST

Ballot submitted as CHALLENGED

CHALLENGE ballots are NOT included in the final tally; they are decrypted and their contents published for the voter to inspect

Ballot Upload

CAST Ballots

CHALLENGE Ballots

Admin and Guardian Devices

Election Setup

Generate Election Context

Upload Election Manifest

Election Key

Manifest.json

Guardian Key Ceremony

Guardian Tally Ceremony

Encrypted Tally

Decrypted Tally

Decrypted CHALLENGE Ballots

Election Record

Verifier

GitHub

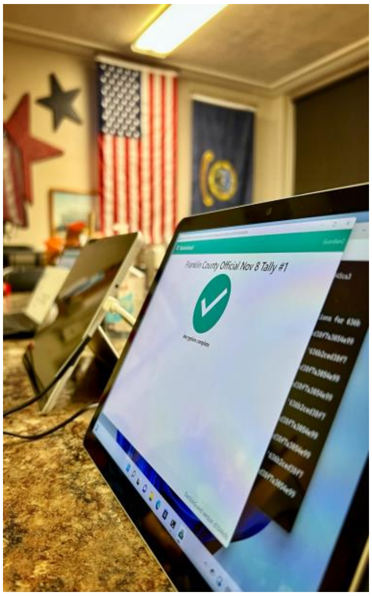
Confirmation Website

Election Record

Microsoft Democracy Forward

ElectionGuard

Increasing confidence in Elections



What is it? ElectionGuard is open-source software that improves confidence and participation in elections. It allows voters to verify that their ballots were included in the published tally, that their selections are correctly recorded by the voting system, and for independent verifiers to confirm the recorded votes are correctly tallied.

How does it work? ElectionGuard runs alongside an existing voting system. It creates a parallel, encrypted copy of every vote, and uses advanced cryptographic techniques and security procedures to preserve the secrecy of the ballot and privacy of every voter. Voters are given a confirmation code with a website URL that they can go to after the election ends to verify that their ballots were included.

ElectionGuard also enables Guardians – respected members of a community chosen by election administrators – to prepare the election and enable the voting system to encrypt ballots. Once the election is over and the encrypted ballots are collected, the Guardians compute a tally without decrypting the ballots. Consistent with modern election voting practices, ceremonies are conducted offline, so no external actors can tamper with the process. Verifiers built by independent third-party organizations such as MITRE can then ensure the recorded votes have been correctly tallied.

2022 Highlights

- **Ran successful pilot in 2022 Franklin County, Idaho General Election**
- **Produced End-to-End Verifiability Report**
- **Gave presentation to the Election Assistance Commission (EAC) on independent verifiability**

Where can I get ElectionGuard? ElectionGuard is available for free on Github. It is being tested and brought to market via a series of diverse and complex pilot elections. It was first tested in Fulton, Wisconsin, in 2019 in partnership with VotingWorks.

More recently, ElectionGuard was used in the 2022 Franklin County, Idaho general election. Hart InterCivic incorporated ElectionGuard into their Verity precinct scaner. The Center for Civic Design helped with messaging and voter feedback to confirm we accomplished our objectives; Enhanced Voting built the public website for voters to check their confirmation codes; and MITRE verified the election was correctly tallied.

We are currently working on ElectionGuard 2.0, which will enable additional voting methods, faster and more efficient ballot encryption, and a production-level version of the software used by Election Administrators.

In 2023, **we are seeking additional pilots to test additional voting methods** such as vote by mail.

For more information about ElectionGuard, please visit our website at: https://www.electionguard.vote or contact ElectionGuard@microsoft.com