DVSorder

Mitigation Tool Technical Details

Published October 14, 2022

DVSorder is a privacy flaw that affects Dominion Voting Systems (DVS) ImageCast Precinct (ICP) and ImageCast Evolution (ICE) ballot scanners, which are used in parts of 21 states. Under some circumstances, the flaw could allow members of the public to identify other peoples' ballots and learn how they voted.

This vulnerability is a privacy flaw and *cannot* directly modify results or change votes. Nevertheless, the secret ballot is an important security mechanism, and some voters —especially the most vulnerable in society—may face real or perceived threats of coercion unless the privacy of their votes is strongly protected.

Many jurisdictions publish data from individual voted ballots, such as cast-vote records (the votes from each ballot) or ballot images (scans of each ballot). This data is usually supposed to be randomly shuffled, to protect voters' privacy. The DVSorder vulnerability makes it possible to unshuffle the ballots and learn the order they were cast. This sometimes makes it possible to determine how specific individuals voted.

Jurisdictions can continue to publish ballot-level data if they take steps to "sanitize" data from vulnerable Dominion scanners. We have created a sanitization tool to help. Public access to election data, including cast-vote records and ballot images, can be valuable for voter confidence, and DVSorder is not a reason to reduce transparency.

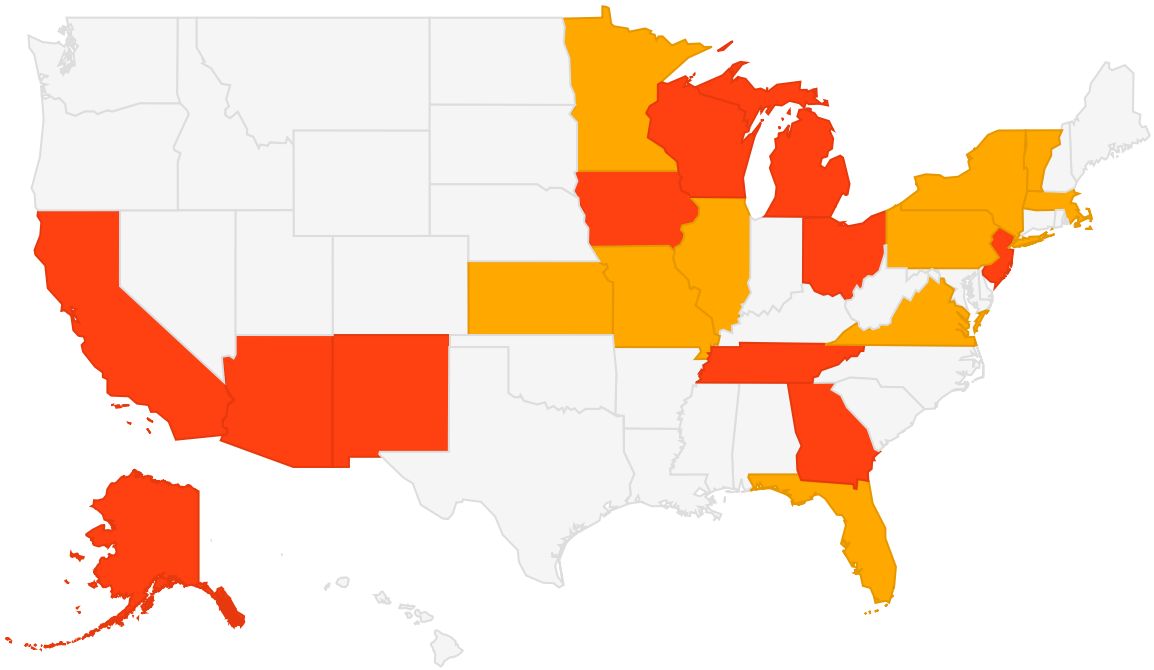
We were able to discover the vulnerability using only publicly available information, and it could potentially be discovered and exploited by anyone, without any access to equipment or breach of controls. Although sanitizing data will protect against exploitation by the public, the original copies of the records remain vulnerable. This means there will still be risks from insiders or data breaches until the scanners are eventually patched. We are making our findings public to ensure all localities are

informed in time to avoid releasing vulnerable data from the November election. We alerted Dominion, CISA, EAC, and state officials prior to publication.

This research was conducted by Braden Crimmins, Dhanya Narayanan, Josiah Walker, and J. Alex Halderman at the University of Michigan and Drew Springall at Auburn University. We can be contacted at **team@DVSorder.org**.

Which jurisdictions are at risk?

According to data from Verified Voting, parts of 21 states and Puerto Rico use the vulnerable Dominion scanners. So far we have identified jurisdictions in 11 states that appear to have published vulnerable data from recent elections: Alaska, Arizona, California, Georgia, Iowa, Michigan, New Jersey, New Mexico, Ohio, Tennessee, and Wisconsin.



Some jurisdictions make this shuffled ballot data public, most commonly in the form of *ballot images* (scans of each individual ballot) or *cast vote records* (data files that record the votes from individual ballots). Dominion's documentation implies that the shuffled data can be safely distributed without compromising voters' privacy, as does information Dominion provided during state equipment purchasing: "*The ballot images are given a random ID number as their file name, and when the images are extracted by the [EMS] application, they are randomized, thus ensuring the ballot images are de-coupled from voter order.*"

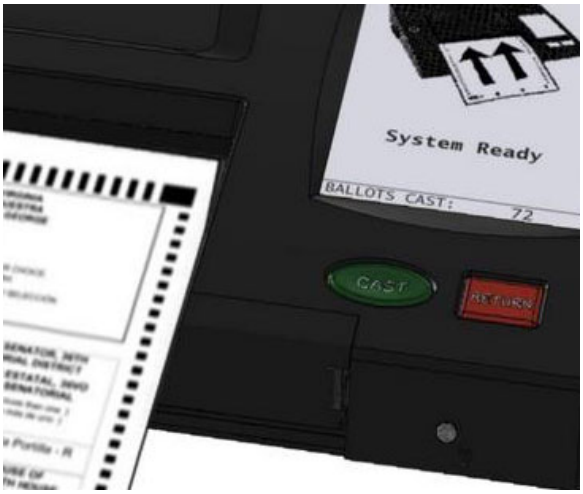
Unfortunately, the Dominion ICP and ICE scanner software is flawed such that ballot record IDs are assigned in a predictable manner. This allows anyone to unshuffle the ballot images or cast vote records and learn the order in which they were cast.

Although the DVSorder vulnerability is specific to two models of Dominion scanners, we recommend that other voting equipment vendors review the technical details and confirm that their implementations do not reveal the order in which ballots were cast.

How does knowing the ballot order threaten privacy?

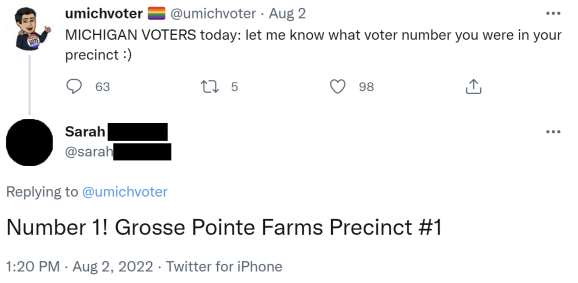
There are several types of scenarios where the DVSorder vulnerability could be exploited to identify how specific people voted:

- In most jurisdictions, scanners display a public counter that shows how many ballots have been cast. Anyone can note the counter value when they vote and thereby learn the ballot sequence numbers of people who vote before and after. For example, suppose a man uses the scanner immediately after his wife. By noting the counter value just before scanning his ballot, the man can later identify his wife's ballot in published cast vote records or ballot image data and see how she voted.

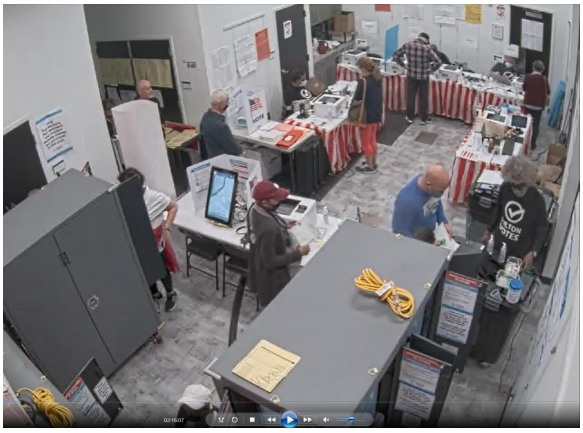


- Poll workers or election observers could similarly note the public counter value to target specific voters. They could also keep a complete record of who uses the scanner, in order, which would allow them to deanonymize all ballots cast at the precinct.

- Some voters publicly disclose their polling places and voter numbers on social media or to others, as in the tweet shown here. As long as the voter has accurately stated their position in the ballot sequence, this would allow anyone to determine how they voted from vulnerable CVRs or ballot images.



- Some localities record all-day surveillance footage inside polling places. (This image is from a day-long video from a county in Georgia and was obtained by others prior to our work via a public records request.) If the jurisdiction releases vulnerable CVRs or ballot images, anyone could associate each ballot with footage of the voter casting it. A larger number of jurisdictions treat voter check-in records or poll books as public records. These can heighten the risks posed by the vulnerability, as they often track the order in which voters receive their ballots, which can match or closely approximate the order of casting.



- Some localities publish scanner log files (`slog.txt` files) from the ICP or ICE. Although these logs by themselves pose little risk to privacy, they can be combined with the DVSorder vulnerability to determine the *exact time* that each CVR or ballot image was cast (subject to the accuracy of the scanner's internal clock). This provides an additional route to identify voters' ballots. As one example, journalists sometimes film or photograph candidates and

other political figures as they vote. Such media is often timestamped and could be used by anyone to deanonymize those individuals' ballots, even long after the election.

What machines and kinds of data are vulnerable?

All versions of the Dominion ICP and ICE for which we have located public ballot-level data appear to be vulnerable to DVSorder, including versions that have been certified by the U.S. Election Assistance Commission (EAC). The problem is specific to the ICP and ICE; ImageCast Central scanners and ImageCast X DREs do not appear to suffer from the flaw. (The ImageCast Central (ICC) intentionally labels ballots in the order they are scanned.)

ImageCast Precinct (ICP)

ImageCast Evolution (ICE)

Dominion's EMS software can export ballot-level data in several forms. Some examples of the most commonly published types of data that may be vulnerable are:

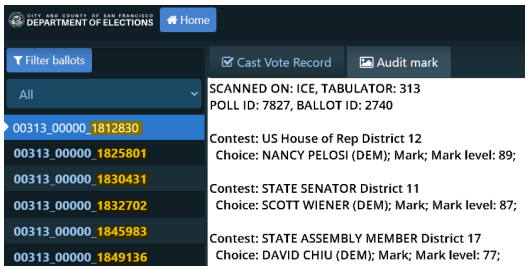

```
  "TabulatorId": 145062,
  "BatchId": 0,
  "RecordId": 180517,
  "CountingGroupId": 2,
  "ImageMask": "D:\\NAS\\20201103 General\\Results\\
\Images\\145062_00000_180517*.*",
  "SessionType": "ScannedVote",
  "VotingSessionIdentifier": "",
  "UniqueVotingIdentifier": "",
  "Original": {
    "PrecinctPortionId": 1166,
    "BallotTypeId": 1076,
```


JSON cast-vote records (CVRs)

TabulatorNum	BatchId	RecordId	CountingGroup	PrecinctPortion
987	0	546984	Election Day	City of Detroit, P
987	0	749347	Election Day	City of Detroit, P
987	0	299063	Election Day	City of Detroit, P
987	0	607574	Election Day	City of Detroit, P
987	0	964848	Election Day	City of Detroit, P
987	0	60943	Election Day	City of Detroit, P
987	0	882503	Election Day	City of Detroit, P
987	0	317204	Election Day	City of Detroit, P
987	0	989938	Election Day	City of Detroit, P
987	0	700677	Election Day	City of Detroit, P

CSV cast-vote records (CVRs)

Ballot image TIF files
(with record IDs in filenames)



Ballot audit website
(with record IDs in filenames)

Only data that represents individual ballots and their record IDs is affected by DVSorder. Summary results such as statements of votes cast (SoVCs), precinct- or scanner-level totals, election-night result reports, and poll tapes are not vulnerable to this privacy flaw.

How can election officials mitigate this?

DVSorder affects only two specific models of ballot scanners: the Dominion ImageCast Precinct (ICP) and ImageCast Evolution (ICE). Jurisdictions that do not use these scanners are unaffected and do not need to take any action. DVSorder should not motivate unaffected jurisdictions to decrease their transparency.

Localities that use the Dominion ICP or ICE can prevent the flaw from being exploited by the public by taking specific steps to “sanitize” ballot-level data before publishing it:

Manually Sanitizing CVRs (CSV format only)

Dominion cast-vote records (CVRs) in CSV format use a simple data scheme that can be sanitized manually. To do so, open the `.csv` file in Excel and delete column D, labeled “RecordId”, then save the file. Removing the record IDs from JSON-format CVRs and ballot image filenames is more labor intensive, so we recommend using our data sanitization tool described below.

Automated Data Sanitization Tool (all formats)

We created an open-source software tool that can automatically reprocess Dominion cast-vote records (CVRs) and ballot image files so that DVSorder can no longer be exploited. The tool can sanitize CVRs in `.csv` or `.zip` format and folders of ballots images in `.tif` format.

Sanitizing published ballot-level data cannot affect official election results, because results are generated directly from the election management system (EMS), not from the ballot-level data released to the public. However, as with any third-party software,

jurisdictions *should not* run our sanitization tool on their EMS computers. Instead, we recommend copying vulnerable CVRs or ballot images to an external system and running the tool there. Our tool is open-source software, and we encourage anyone interested to view the code and test its behavior.

More about our tool:
• Read the documentation
• View the source code

Election officials who need assistance can contact us, and we will be happy to provide any help we can.

Is there a software patch?

Sanitizing ballot-level data before publishing it makes the data just as safe to release as if the DVSorder vulnerability did not exist. However, even if jurisdictions sanitize the data they make public (or if they do not publish any ballot-level data), the flaw still carries risks. For instance, unsanitized data could be stolen in a data breach or accessed by malicious insiders, who could exploit the flaw to learn how people voted.

Completely mitigating these risks will require Dominion to change the ICP and ICE firmware to use a secure method of generating ballot IDs. The U.S. Election Assistance Commission (EAC) has informed us that Dominion plans to correct the flaw in future firmware versions. However, our understanding is that no patches will be available until after the November election, at least for federally certified versions of Dominion systems. Election officials should contact Dominion for further information and to inquire as to patch availability.

What disclosure was made prior to publication?

We notified Dominion about the vulnerability on August 23, 2022. Our disclosure letter to Dominion informed them that we planned to publish information about the flaw as soon as 30 days later and offered to assist them in understanding and mitigating the problem. The company acknowledged receipt of the disclosure on August 29, but we have not received any subsequent communication from them. We informed the U.S. Election Assistance Commission (EAC) and the Cybersecurity & Infrastructure Security Agency (CISA) about the vulnerability on September 2.

Two weeks after we notified Dominion, it sent a "customer notification" to jurisdictions that use the ICP and ICE. (Dominion did not provide us a copy, but we

obtained one from an affected jurisdiction.) While the notification appears to be in response to our disclosure, it **does not mention** that the scanners have a vulnerability that reveals the order in which ballots were cast. Instead, it directs election officials to "*follow any state or local requirements guiding public access to and release of cast vote records*" and to "*consult their legal advisors for guidance on how best to ensure that [voter secrecy] protections are applied, particularly if simultaneously releasing any record (i.e. [sic] video) that could reveal a voter's identity in the order in which they cast their ballot.*"

We observe that such legal advisors would likely rely on Dominion's prior, inaccurate representations that ballot-level data is appropriately randomized to protect privacy. By failing to provide information about the specific risks posted by the DVSorder flaw, Dominion's notice appears to have left jurisdictions unable to make informed decisions about whether and how to release election data.

Before publication of this website, we sent our own notifications to the state election directors in states that we believe use ICP or ICE scanners.

Why are you publishing this before the election?

We consulted with other experts and considered a range of equities before concluding that the public interest would be best served by publishing now.

The vulnerability is unusual in that it doesn't require exotic skills or special access to discover or exploit, but rather only publicly available information. This means there is an appreciable risk that malicious parties would independently discover the flaw, or that they already have. With the bar so low, we're concerned that people will attempt to exploit it following the midterms.

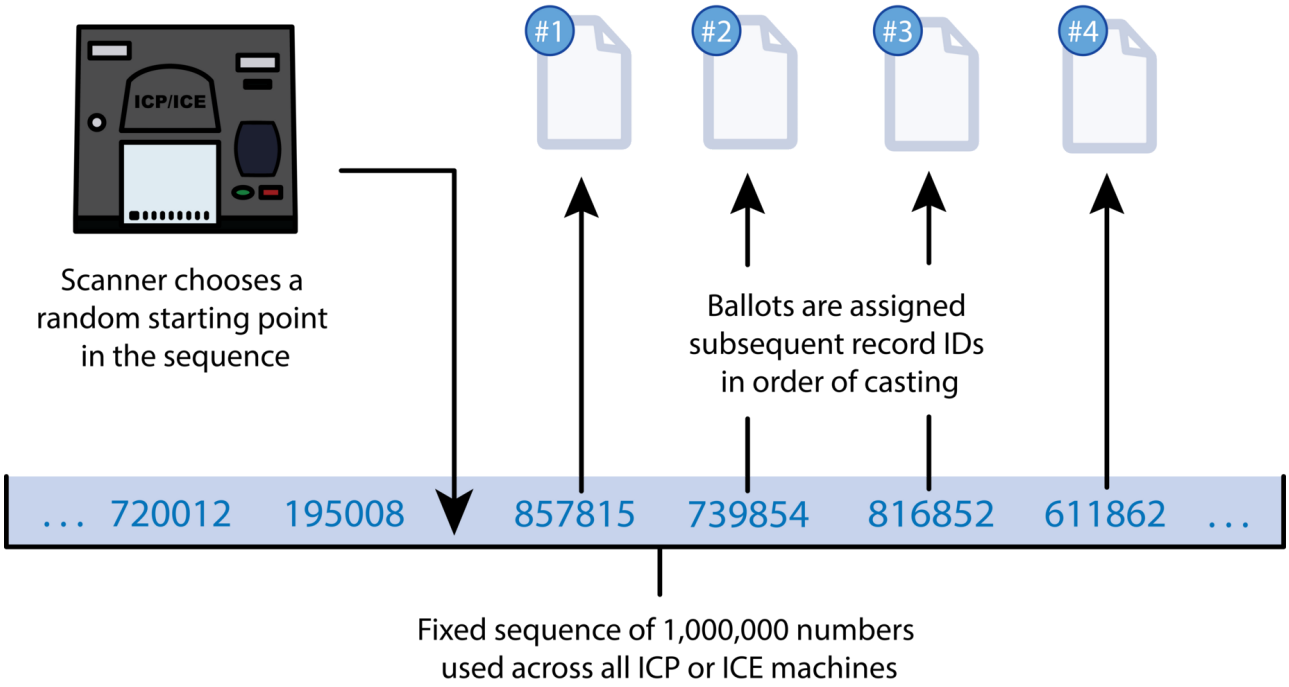
If we did not make our findings public before the election, jurisdictions would almost certainly publish a large volume of vulnerable data in November. Once released, this data would remain vulnerable in perpetuity, even if the scanners themselves were later patched. Raising the alert now gives election officials time to respond effectively. Our priority is to prevent this flaw from affecting voters in the midterms, which is ultimately the best way to uphold public trust.

Technical details

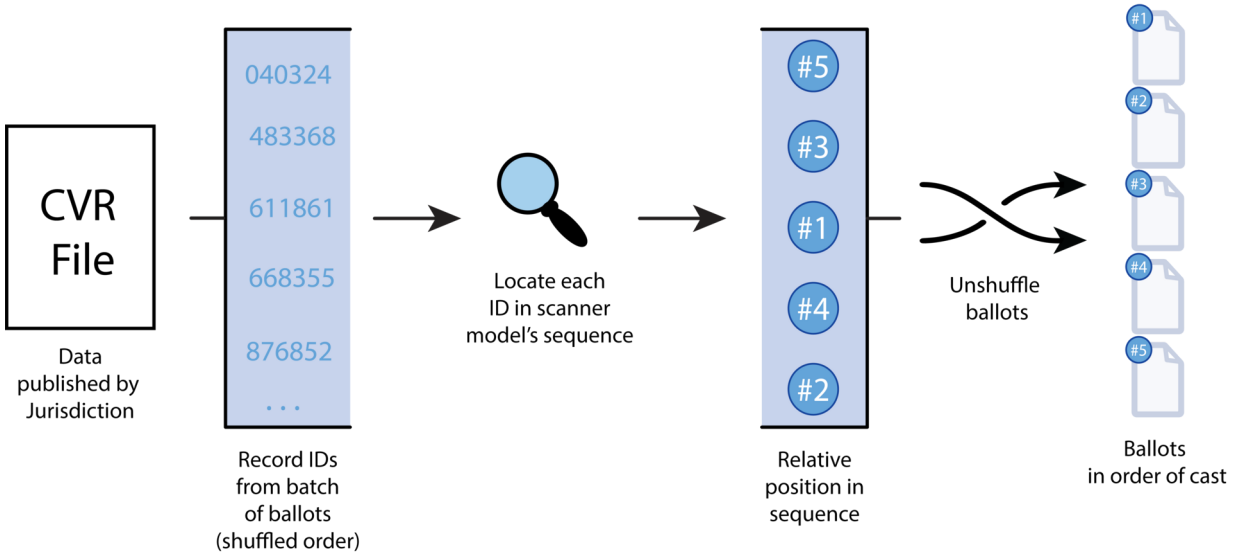
The Dominion ICP and ICE generate ballot record IDs using a pseudorandom number generator (PRNG). The PRNG they use is based on a linear congruential generator

(LCG). LCGs have long been known to be unsuitable for most security applications, both because the sequences they generate have obvious patterns and because their entire output is predictable given only a few samples. Dominion attempts to obfuscate the LCG output using some simple transformations (which differ slightly between the ICP and ICE), but these are insufficient to make the PRNG secure.

The ICP and ICE PRNGs each output a fixed sequence of 1,000,000 numbers (a permutation of the numbers 0-999,999) that is the same across all devices of each model. For a given batch of ballots, only the *starting point* within the sequence is randomized. The ballot record IDs are simply consecutive values in the fixed sequence from that point forward:



If an attacker knows the record IDs from the ballots in a batch (from CVRs, ballot image filenames, or any other source), they merely need to locate them in the PRNG output sequence for the scanner model. The record ID that appears first in the sequence corresponds to the earliest ballot, and all other record IDs will appear following it in the sequence, in the order in which they were cast:



We identified the vulnerability from just a short series of record IDs in voted order, which we obtained from publicly available data. Even in a small sample (like the example shown below), there are clear repeating patterns in several of the digit positions. This immediately suggests the use of a simple, non-cryptographic PRNG, such as an LCG:

303001	907271	801991
720012	224222	722982
195008	599278	693998
857815	956625	858485
739854	332644	435414
611861	513631	617401
876852	170642	074412
483368	385138	481708
668355	764145	266715
040324	149184	042754

Starting from this observation, multiple members of the team were able to independently reconstruct the complete PRNG algorithm within a few days.

Both the ICP and ICE PRNGs generate record IDs through a simple sequence of steps. They start with the LCG $x_{n+1} = x_n + 864{,}803 \bmod 1{,}000{,}000$. The output is then obfuscated by a simple substitution cipher in which the digits [0,1,2,3,4,5,6,7,8,9] are replaced by [5,0,8,3,2,6,1,9,4,7]. The digits are then reordered following a fixed permutation that is different on the ICP and the ICE.

This code reproduces the complete record ID sequence for each scanner model:

```
def generate_sequence(p):
    return [sum([5,0,8,3,2,6,1,9,4,7][864803*n//10**p[i]%10]*10**i for i in range(6
        for n in range(1000000)]
icp_sequence = generate_sequence([2,3,1,5,0,4])
ice_sequence = generate_sequence([1,5,0,4,2,3])
```

dvs_prng.py hosted with ❤️ by GitHub view raw

To validate and test for the vulnerability, we created a **proof-of-concept implementation**. This program inputs a CSV- or JSON-format CVR file and outputs the fraction of ballots that appear to be vulnerable.

We will provide additional technical details in a forthcoming research paper.
