Online Voting

Consensus among people
(machines can be used where no coersion present)

KWLUG December 2020 talk
(hashed out on mailing list)

Narratives: exhibit 1

[kwlug-disc] Say No To Electronic Voting ...

Narratives: exhibit 2

Something tells me that with the zero-knowledge systems and recent inventions like zerocash and zerocoin, this statement is not actually true anymore.

Sadly I'm not smart enough to know for sure.

Decisions amidst uncertainty?

Year	Municipal Elections held online in Ontario
2003	12
2006	20
2010	44
2014	97
2018	150

source: https://en.wikipedia.org/wiki/Electronic_voting_in_Canada#Ontario_2

Clean Narrative

	Paper ballot	Online voting
Integrity of process	Physics helps	Needs Verification (no Physics help here)
Coersion threat	Mixing of similar ballots prevents checking of coersion compliance	Verification checks coersion compliance

Clean Narrative: contrast

Key difference, obvious and, thus, unspoken

	Paper ballot	Online voting
Integrity of process	Physics helps	Needs Verification (no Physics help here)
Coersion threat	Mixing of similar ballots prevents checking of coersion compliance	Verification checks coersion compliance

No Verification => tech attack

Without end-to-end Verification,
attack integrity of servers, client software

	Paper ballot	Online voting
Integrity of process	Physics helps	Needs Verification (no Physics help here)
Coersion threat	Mixing of similar ballots prevents checking of coersion compliance	Verification checks coersion compliance

Verification => human attack

Bandits use verification to check how victim votes, completing coersion process

	Paper ballot	Online voting
Integrity of process	Physics helps	Needs Verification (no Physics help here)
Coersion threat	Mixing of similar ballots prevents checking of coersion compliance	Verification checks coersion compliance

Irreconcilable tension of Verification

	Paper ballot	Online voting
Integrity of process	Physics helps	Needs Verification (no Physics help here)
Coersion threat	Mixing of similar ballots prevents checking of coersion compliance	Verification checks coersion compliance

Verification oracle coersion attack

Verification in election means that voter can verify that vote is correctly recorded. Voting system becomes an oracle that answers some question, and depending, on the answer you can tell that vote is correctly recorded.

Let's combine human coercion setting with oracle:

1) Coercer has access to voter's voting material, ids, keys, etc.

2) Coercer uses voting material together with expected, coerced vote values to form a request to an oracle.

3) Coercer checks oracle's reply. Reply indicates if voter put expected vote, or not. Thus, coercer checks if victim did "the right thing", closing coercion loop of information, i.e. verification by an attacker.

Online voting is simple

- Distributed
- End-to-end verifiable

Two process phases

- Registration:

  one person – one ballot

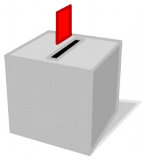
- Voting:

  one person – one vote

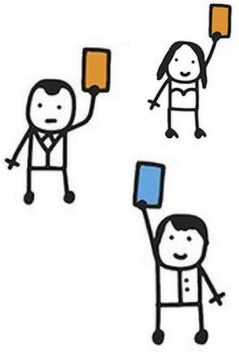
Process participants

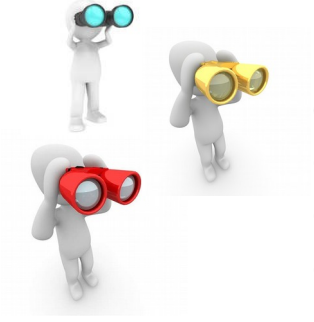
Registrator

Ballot box



Voters





Observers:

- independent organizations
- parties
- citizens

Registration of ballot keys

Registrator

Voter

Registrator has key RegK for signing registration certificates

Authorization

Start of registration session

EK(pub) & VK(pub)
to bind with ballot number (B#)

Certificate binding voter's idenitification with B#, EK(pub) и VK(pub).
Certificate is signed with RegK, which public part is known to everyone.

Application generates following key pairs:

EK – for entry and signing voted ballot

VK – for encrypting ballot

Private keys are kept secret, while public keys are registered with particular ballot number B#

Publication and verification of registrations

Registrator

Publication of triplets
B#, EK(pub), VK(pub)

Certificate from a previous step is not published, and is kept by voter.

anonymous connections

Observers record what ballots are expected and with what keys

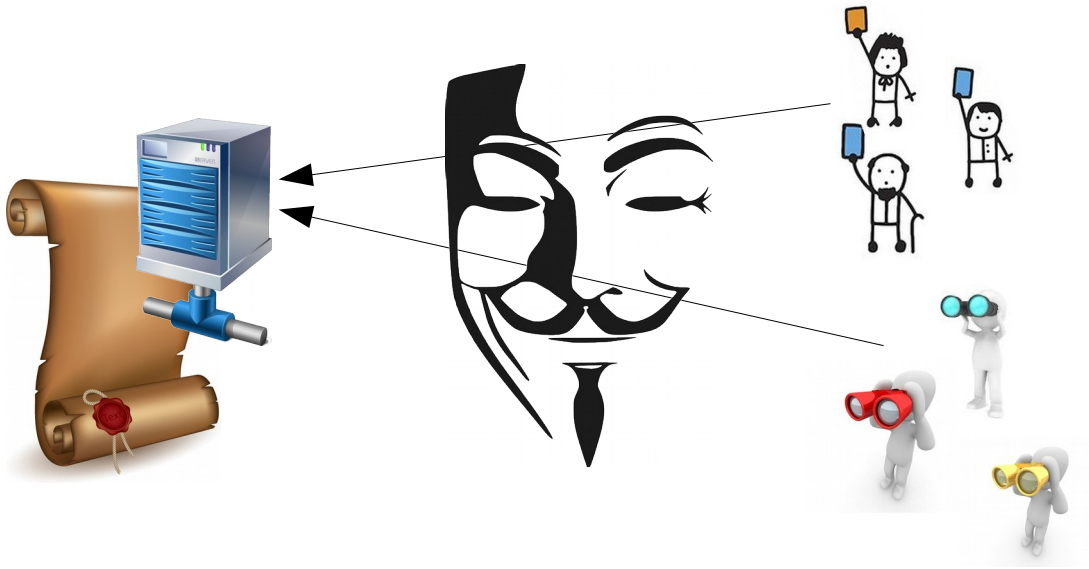
Voters check if their keys correctly registered with their ballot number

Voter uses certificate to prove incorrectness of registration

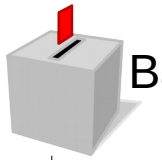
Voting, part 1: publication before voting

Publication:

- Public key of keypair MainK, to which votes are encrypted before counting
- Questions with possible answers (JSON format)
- Formulars to count (JavaScript language)



Voting, part 2: casting votes



Ballot box/server

Voter

App on voter's device shows questions and collects answers.

Authentication with EK for casting ballot B# with answers to questions

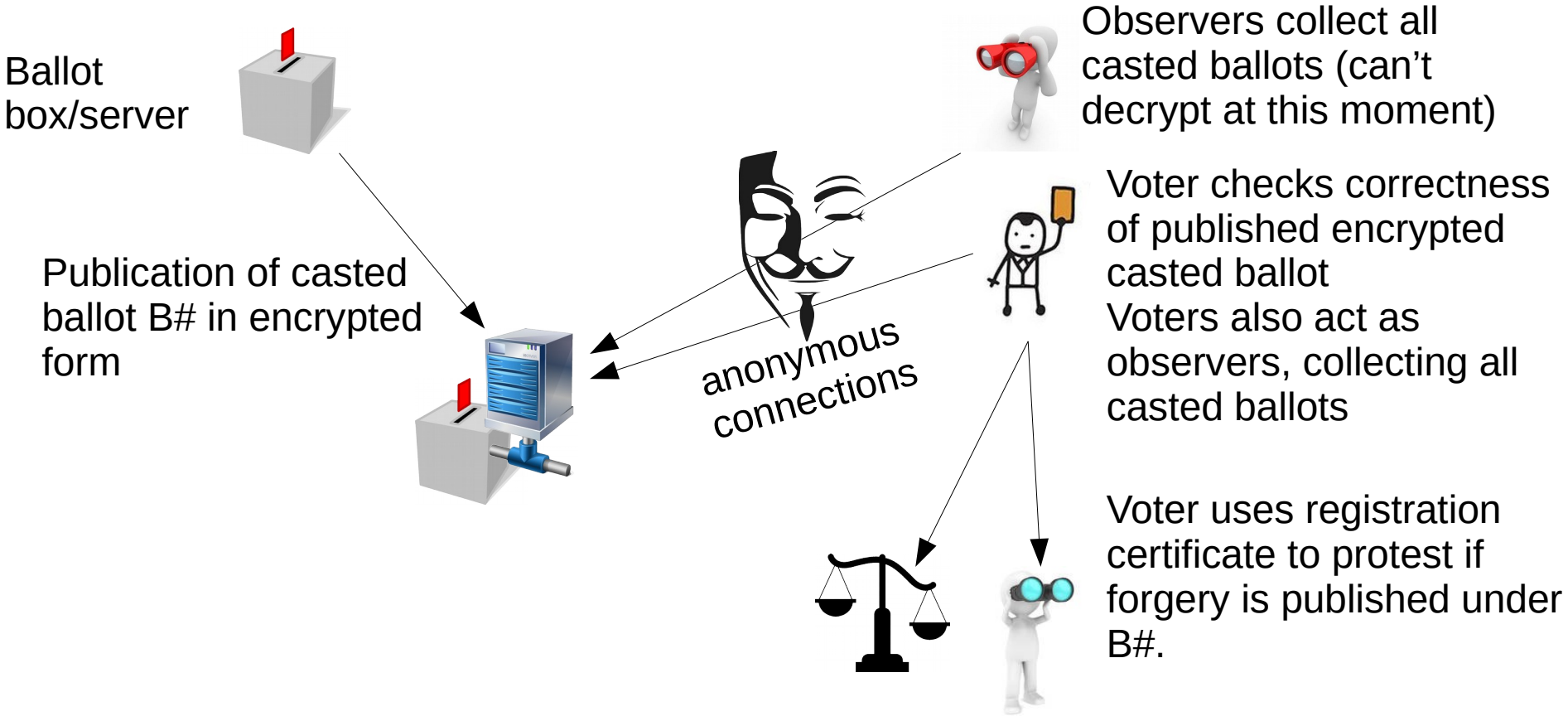
Ballot with answers is encrypted to key pair VK(priv) & MainK(pub).

Open session for B#

VK(pub) has already been published. When private part of MainK is published, everyone will be able to decrypt it with VK(pub) & MainK(priv).

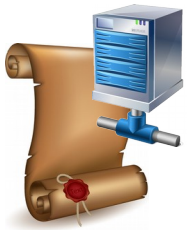
Deposit ballot B# with choices, encrypted to key pair VK & MainK

Voting, part 3: publication & verification of casted encrypted votes



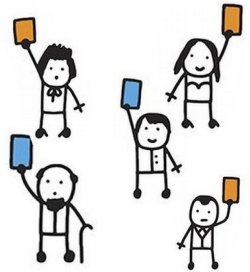
Voting, part 4: publication of Main Key to decrypt and count votes

Access to MainK(priv)

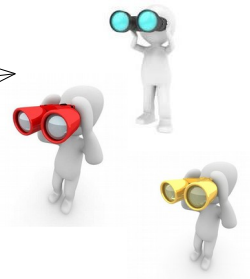


Access to VK(pub) for each B#

Access to all casted B#



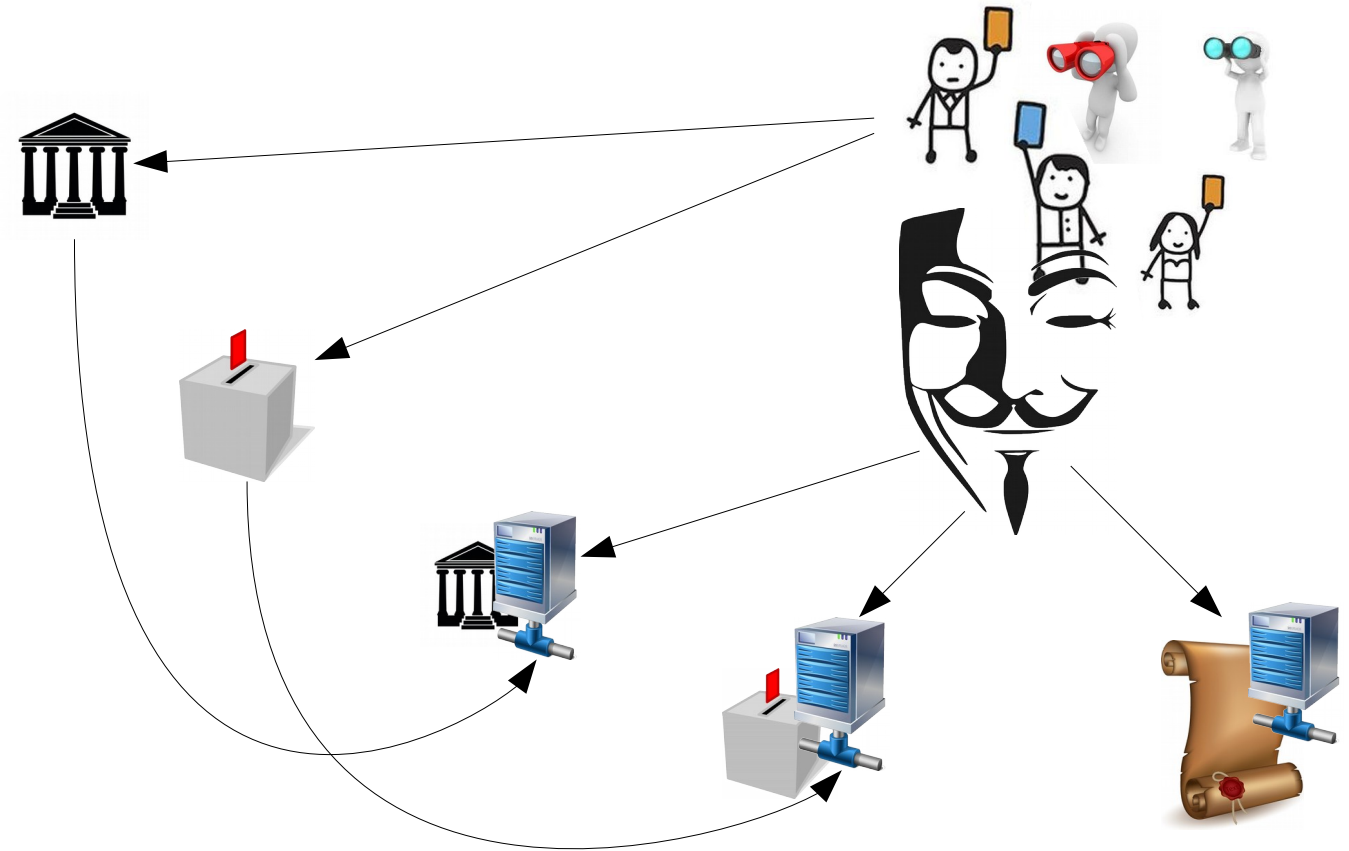
Everyone has access to key MainK(priv) and VK(pub) keys for each casted ballot B#.



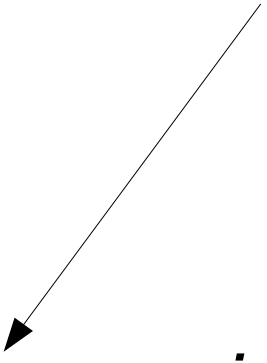
Key pair MainK(priv) & VK(pub) decrypts respective ballot.

Every devices decrypts all ballots and tallies results.

Flows

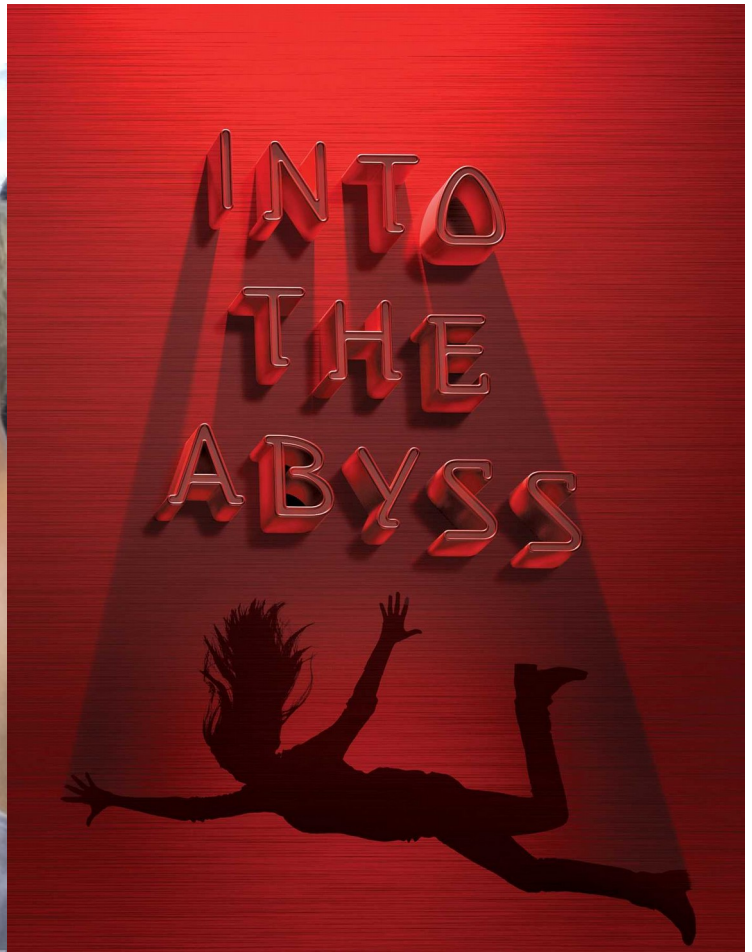


Online voting is simple, ... but



- Distributed
- End-to-end verifiable *also allows coersion*

Elections in Belarus 2020



Belarus 2020: what worked

Wearing wrist band to show everyone your protest vote

Folding ballot with your protest vote

Ensuring that majority knows what majority wants