NTNU

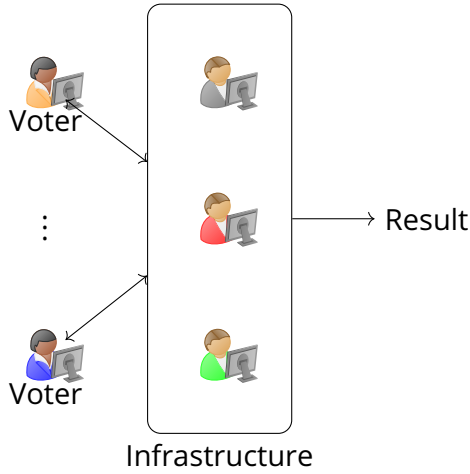
Norwegian University of Science and Technology

Long lasting privacy for e-voting

Oskar Goldhahn

October 25, 2024

Voting



Desired Properties

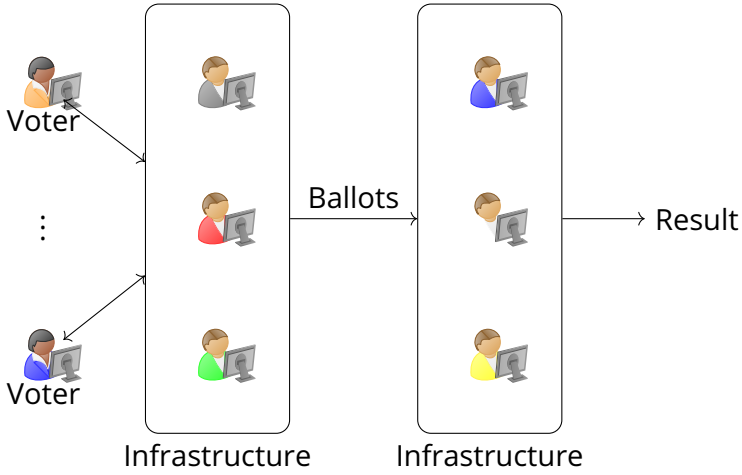
- Integrity (Informal):
  We can gain confidence that the result is what it should be
- Privacy (Informal):
  We do not leak information about what people voted
- Everlasting Privacy (Informal):
  Same as above, but no computational limitations

What's wrong with everlasting privacy?

- Some schemes require anonymous channels[1]
- Some schemes have strong trust assumptions[1]
- Some schemes require trustees to aid the casting process[1]
- Schemes need to be built with it in mind

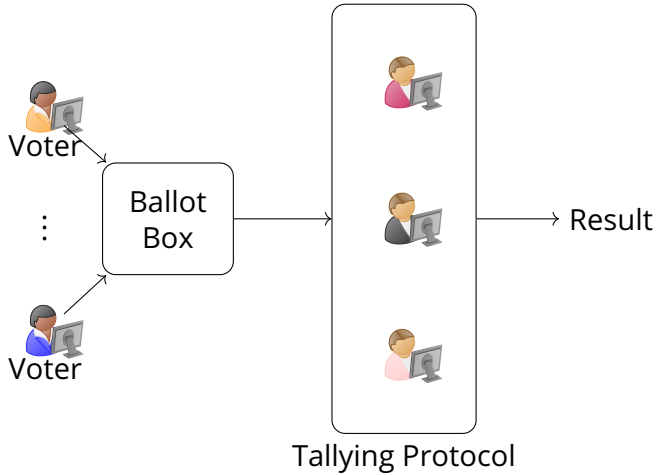
Our Proposal

Strengthen security by tacking existing schemes onto each other, making the initial ballots layered like an onion.

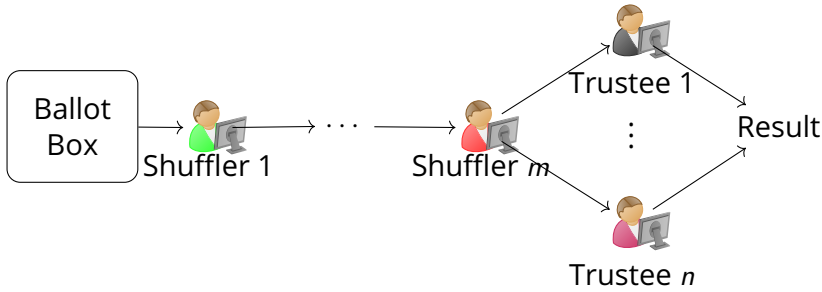


Model

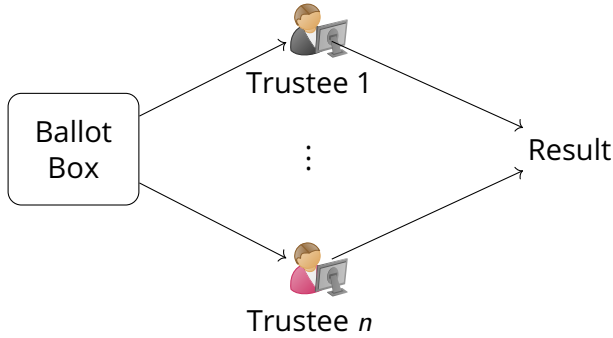
For flexibility we treat tallying as a protocol



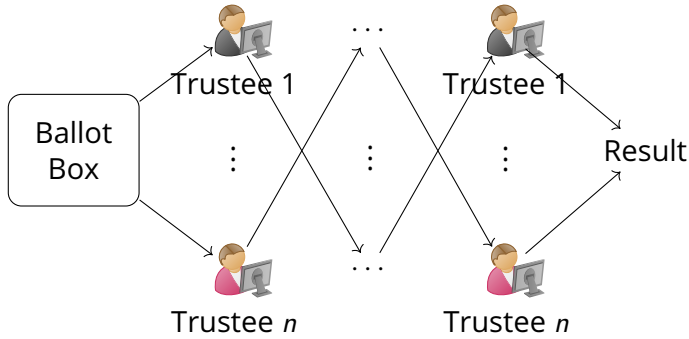
Voting with Mixnets[2]



Voting with Homomorphic Encryption[3, 4]



Voting with generic universally verifiable MPC



NTNU

Security notions

Definition

A Voting Scheme has *Integrity* for a given assignment of honest trustees if an adversary controlling the network and some of the voters cannot produce a result inconsistent with the votes of the honest voters

Security notions

Definition
A Voting Scheme has *Integrity* for a given assignment of honest trustees if an adversary controlling the network and some of the voters cannot produce a result inconsistent with the votes of the honest voters

Definition
A Voting Scheme has *privacy* if an adversary cannot distinguish between tallies of differently permuted honest votes

NTNU

Timing Challenges

Some schemes can produce a result without participation of all trustees

Solution: Trustees should perform both tally protocols concurrently to allow catching up

Timing Challenges

Some schemes can produce a result without participation of all trustees

Solution: Trustees should perform both tally protocols concurrently to allow catching up

---

There is no clear split in time between the two protocols

Solution: The adversary gets to decide when the ballots are handed out to each trustee

Non-uniqueness Challenges

Some schemes permit trustees to swap out malicious votes during the tally

Solution: Use integrity definition that does not guarantee a unique result

Non-uniqueness Challenges

Some schemes permit trustees to swap out malicious votes during the tally

Solution: Use integrity definition that does not guarantee a unique result

---

Malicious trustees might make other trustees disagree on which ballots they should tally in the second protocol

Solution: Let the adversary swap out or remove malicious ballots before handing them to the trustees in the privacy game

Non-determinism Challenge

Encryption does not preserve equivalence up to permutation

Solution: We use randomized algorithms producing votes rather than votes directly in the privacy game

Integrity of Composition

Theorem

The composition of two voting schemes with integrity has integrity

NTNU

Simple Attacks on Privacy

- If the outer protocol forgets some honest ballots, the result reveals information.
- If the adversary can break the outer encryption of an unpublished ballot, they might be able to copy the inner ballot and cast it as their own, which can be detected in the result.

Privacy of Composition

Theorem

The composition of two voting schemes where the first has privacy has privacy

Theorem

If there is a privacy attack on the composition one of the following must be true:

1. *There was a party capable of attacking integrity of the first scheme at the time of the tally*
2. *There is a privacy attack against the second scheme*
3. *There was a party capable of attacking privacy of the first scheme at the time of the tally*

We can strengthen (3.) by adding a layer of a treshold decryption capable encryption scheme where the keys are revealed after the tally

NTNU

Thank you!

Questions?

[1] Thomas Haines et al. “SoK: Secure E-Voting with Everlasting Privacy”. In: *Proceedings on Privacy Enhancing Technologies* 2023.1 (Jan. 2023), pp. 279–293. DOI: 10.56553/popets-2023-0017.

[2] Diego F. Aranha et al. “Verifiable Mix-Nets and Distributed Decryption for Voting from Lattice-Based Assumptions”. In: *ACM CCS 2023: 30th Conference on Computer and Communications Security*. Ed. by Weizhi Meng et al. Copenhagen, Denmark: ACM Press, Nov. 2023, pp. 1467–1481. DOI: 10.1145/3576915.3616683.

[3] Xavier Boyen, Thomas Haines, and Johannes Muller. “Epoque: Practical End-to-End Verifiable Post-Quantum-Secure E-Voting”. In: *2021 IEEE European Symposium on Security and Privacy (EuroS&P)*. Vienna, Austria: IEEE, Sept. 2021, pp. 272–291. ISBN: 978-1-66541-491-3. DOI: 10.1109/EuroSP51992.2021.00027.

[4] Ian Black et al. *Practical Quantum-Safe Voting from Lattices, Extended*. Cryptology ePrint Archive, Report 2022/1686. 2022. URL: https://eprint.iacr.org/2022/1686.

[5] Xavier Boyen, Thomas Haines, and Johannes Müller. “A Verifiable and Practical Lattice-Based Decryption Mix Net with External Auditing”. In: *ESORICS 2020: 25th European Symposium on Research in Computer Security, Part II*. Ed. by Liqun Chen et al. Vol. 12309. Lecture Notes in Computer Science. Guildford, UK: Springer, Cham, Switzerland, Sept. 2020, pp. 336–356. DOI: 10.1007/978-3-030-59013-0_17.

NTNU

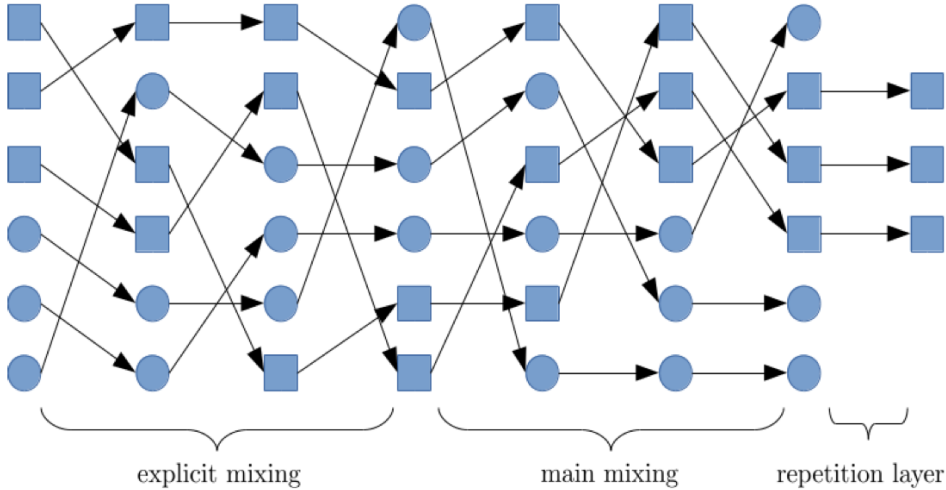
Extra slides

Concrete Construction for Post-Quantum Shuffle[5]

We use a decryption mixnet with trip wires

ballots: $b = \mathbf{Enc}_1(\ldots \mathbf{Enc}_n(\mathbf{Enc'}_1(\ldots \mathbf{Enc'}_n(\mathbf{Enc''}(v))\ldots))\ldots)$

Concrete Construction for Post-Quantum Shuffle[5]



From “A Verifiable and Practical Lattice-Based Decryption Mix Net with External Auditing”

- Lighweight
- Can reuse classical schemes for (non-permutation) privacy
- Voters can object without revealing their vote

- Integrity against fully malicious trustees requires voter participation

Concrete Tally Model

Tally Participant

- (Initial Step): Receives the contents of the ballot box and sends a list of messages
- (Step): Receives a message and sends a list of messages

Network

- (Send): Receives a list of messages from a participant
- (Process): Outputs a message together with a recipent id or terminates with a list of messages

Correctness

Without restrictions the network can do annoying things such as dropping messages or producing bogus messages indefinitely

We want to prohibit this while keeping our assumptions minimal

Definition
A Voting Scheme is *Correct* if newly cast ballots are valid and tallying them using a network that eventually sends every message received if it eventually stops receiving any produces a valid result that is the same as what we would get from counting the votes in plain