Stellot

Secure voter-to-voter internet voting

Stanislaw Baranski

24.03.2023

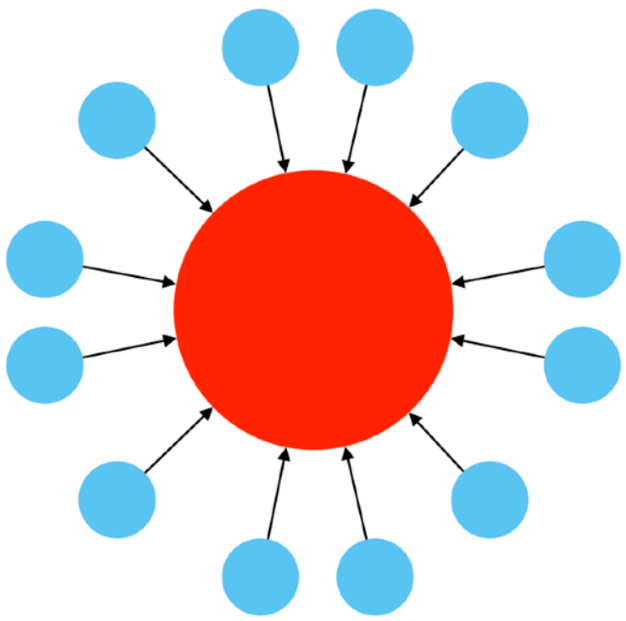
Aim

- We **don’t** want to build
  - A large-scale voting system for presidential elections;
  - A voting system for crypto space only;
  - A perfectly secure, coercion-resistant, protocol.
- Rather, we **do** want to build
  - A voting protocol for small to medium size voting like 100 voters;
  - A voting protocol for people, without a crypto background;
  - A decentralised and secure enough protocol that works.

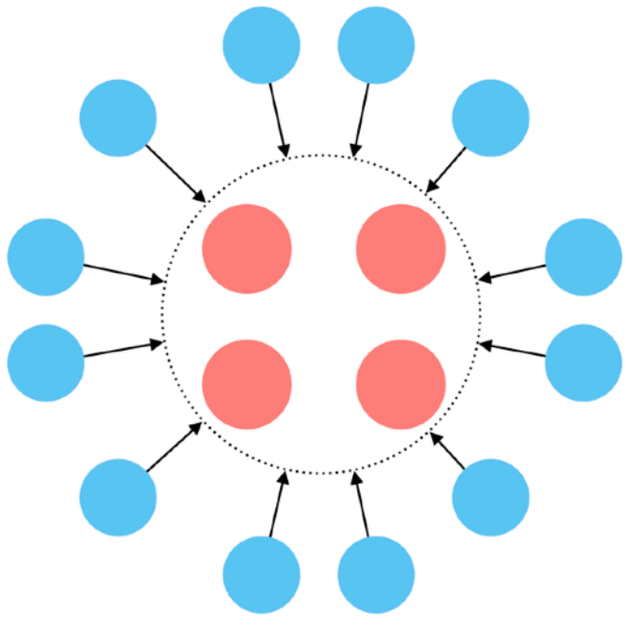
Voter-to-voter trust model

- Most people think about voting in terms of presidential elections. However, voting is used also in small, local votings like housing associations, board members, contests, and all forms of committees.

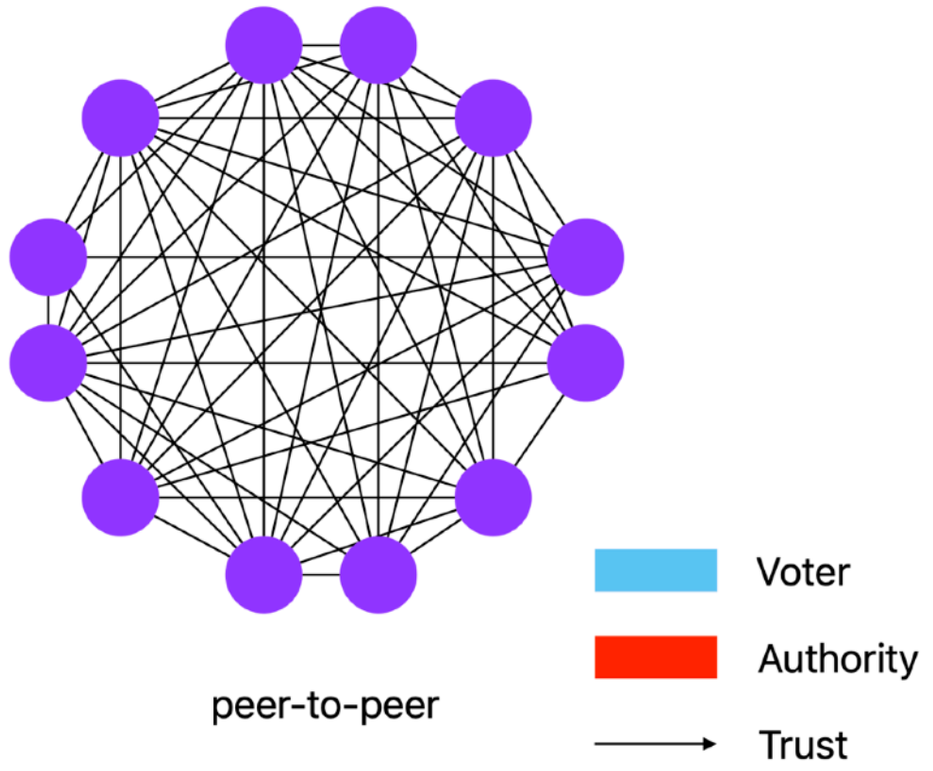
- We want to go even further and conduct the voting on voters' end devices (PC, laptops, or even smartphones) using both blockchain and MPC.



Trusted third party



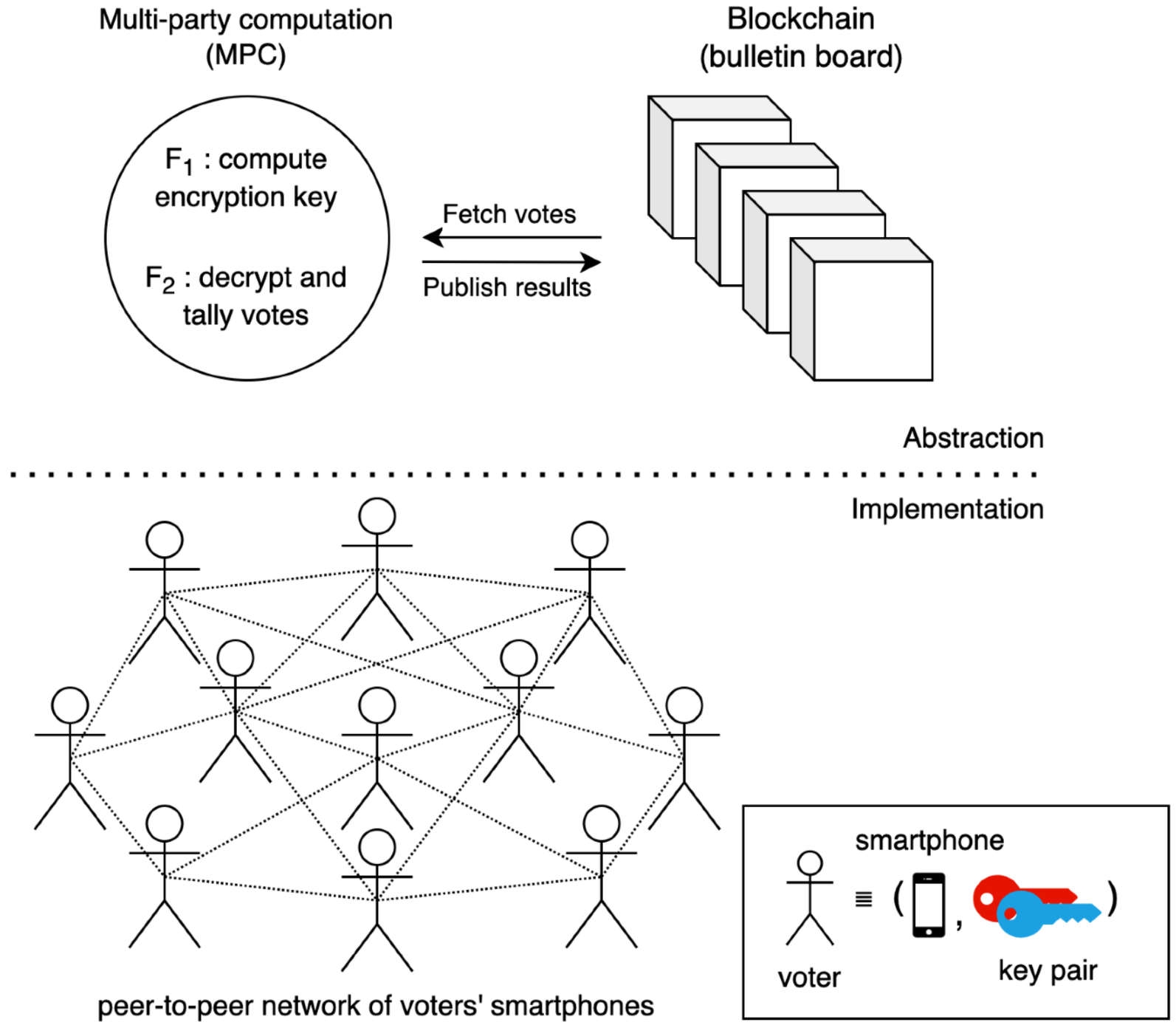
Distributed third party



peer-to-peer

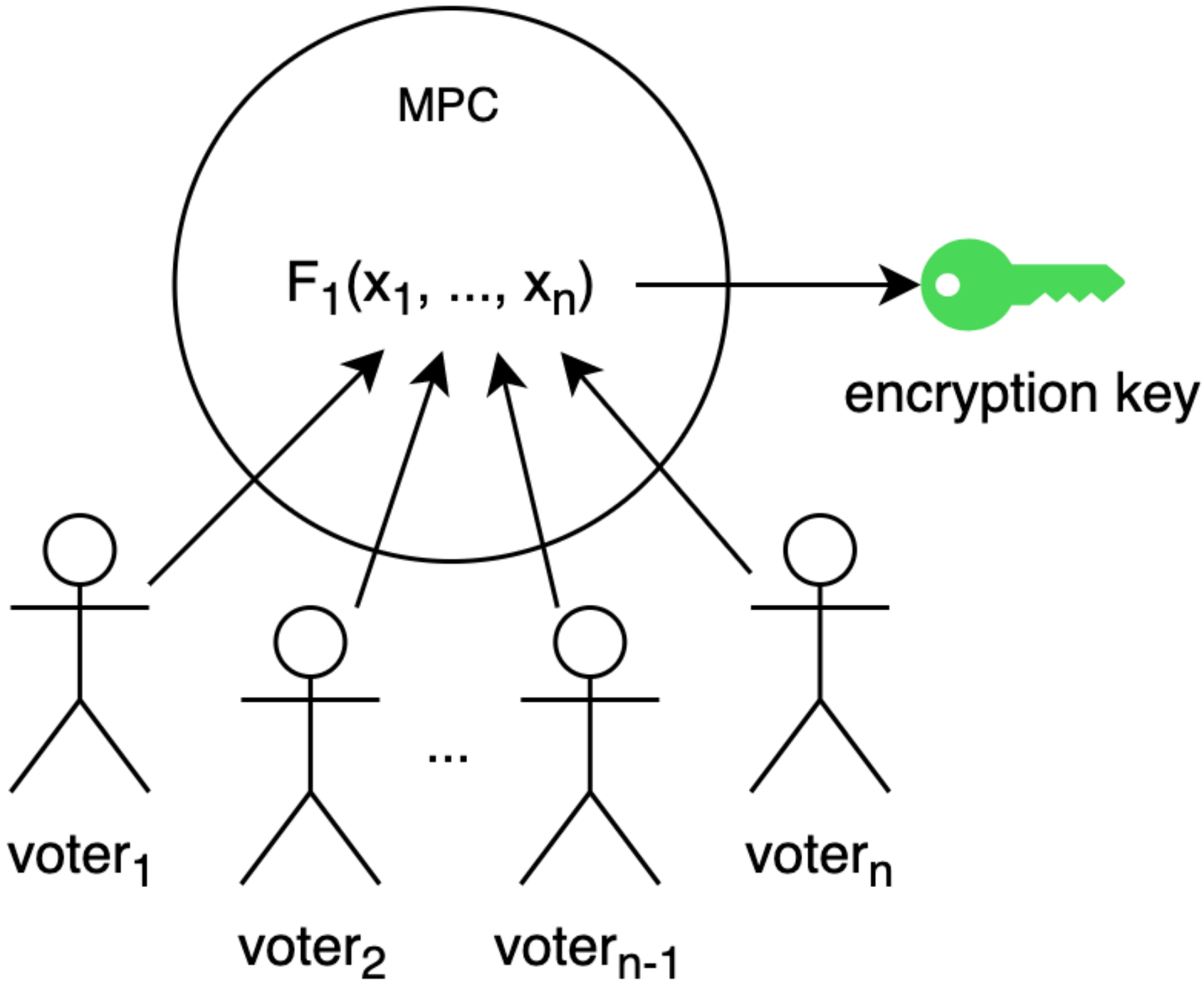
Technical Vision

Architecture



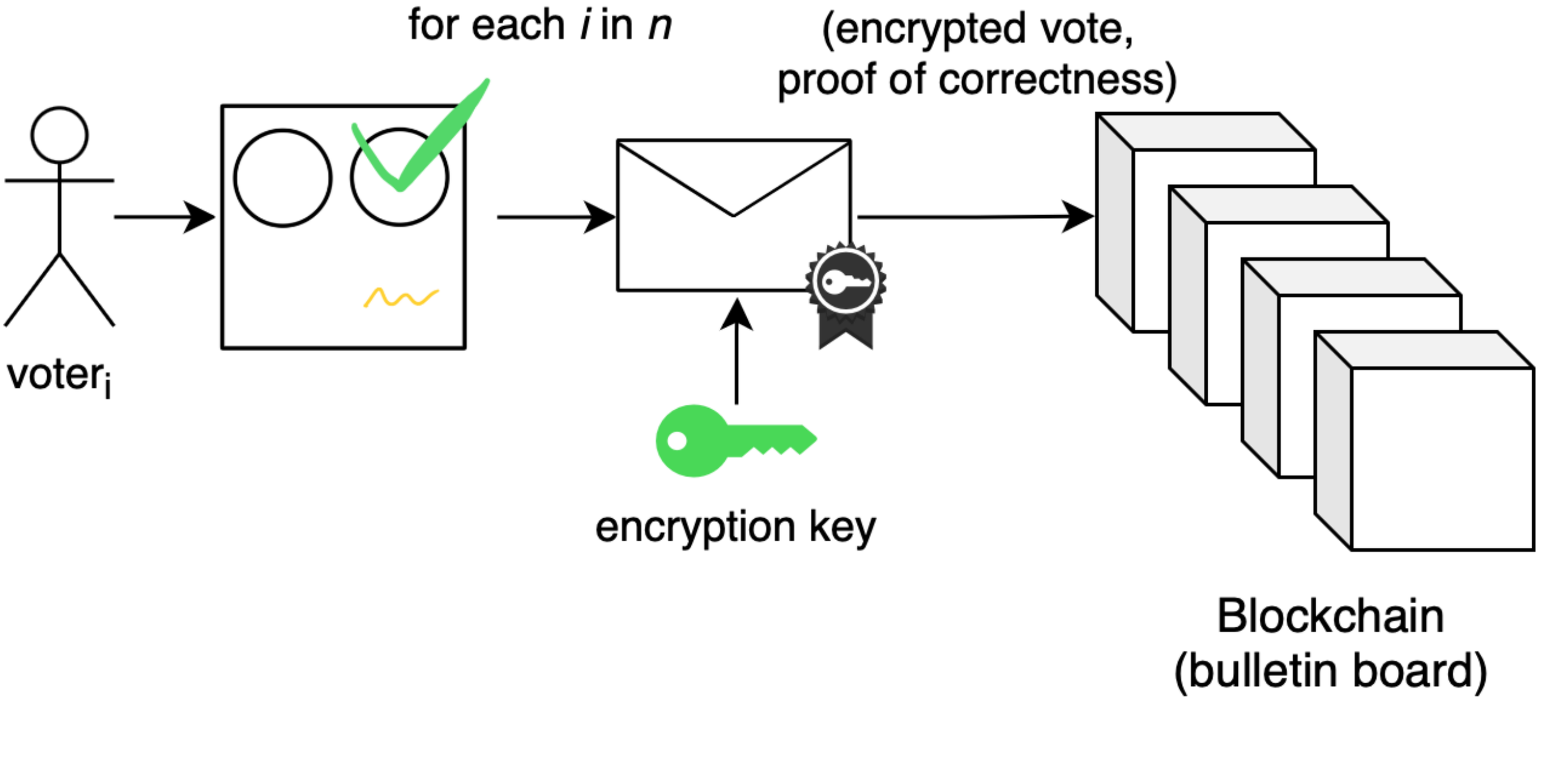
Technical Vision

Setup



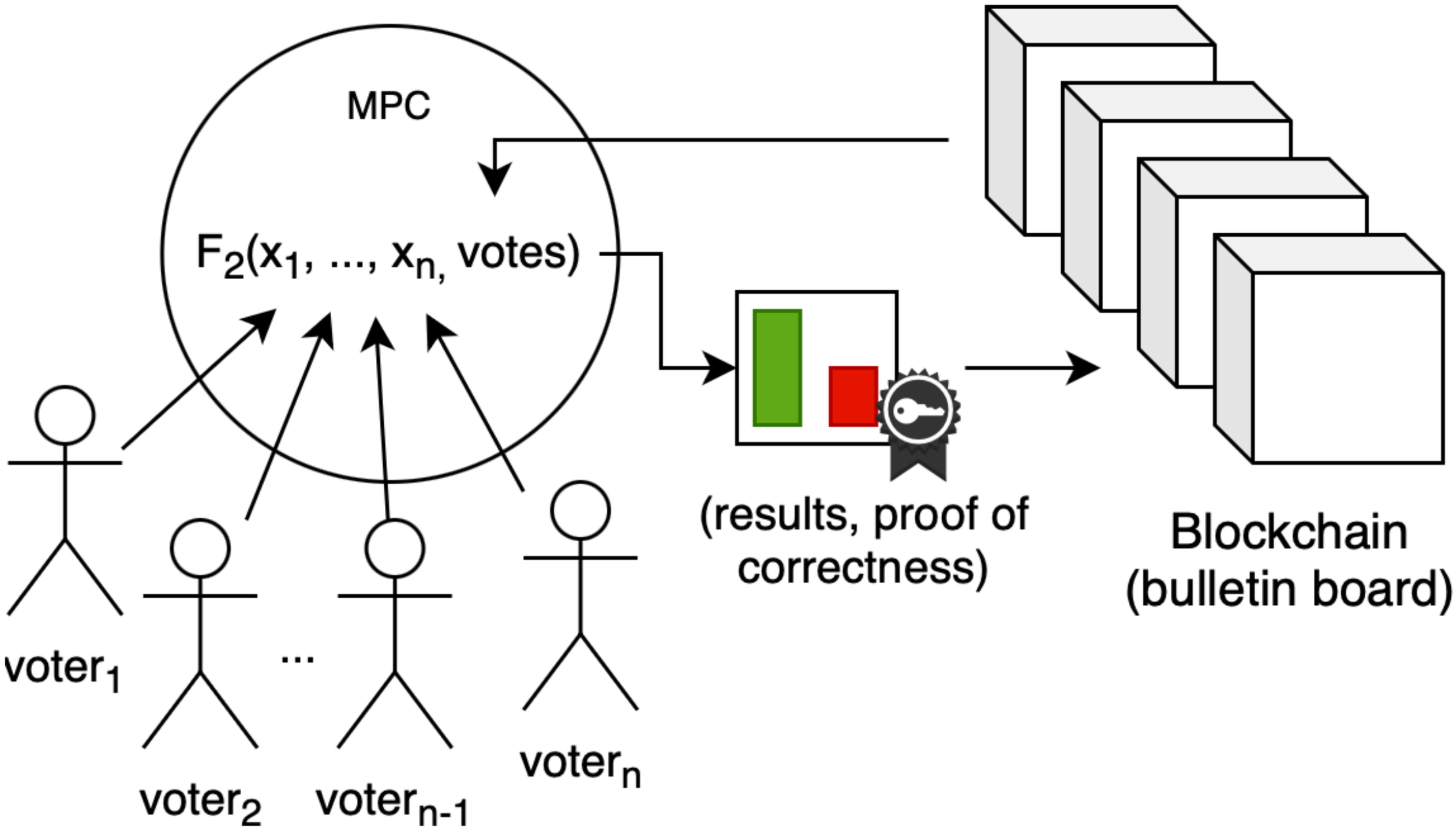
Technical Vision

Voting



Technical Vision

Tally



Results

- Set goals for the protocol:
  - voter-2-voter setting, no trusted authorities, optional participation
  - Asynchronous protocol, participants do not need to be online at any time, they can show up, send a message, and leave.
  - blockchain agnostic, we use blockchain just for a message board—common knowledge of registered votes.

Results

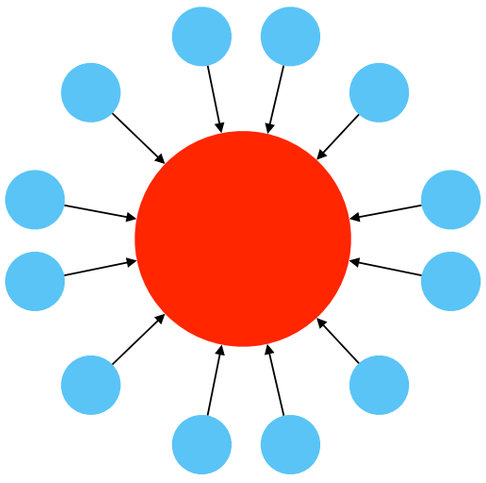
- We have specified our use-case — small-scale, boardroom, housing communities elections.
- Planned the Development roadmap
- Designed the Federated Protocol and backup Registration Protocol.

Property/System using	Trusted server/BC(s) (Polys or ElectionGuard)	Public blockchain	voter2voter network
Transaction fees	No 🌕	Yes 🌑	No 🌕
Running software costs	Yes 🌑	No 🌕	No 🌕
Trust to [1]	Authorities [2] 🌗	Miners 🌗	Voters 🌗

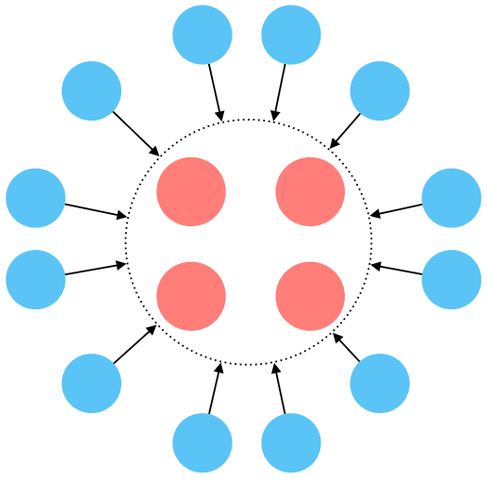
Unsolved problems

- In-browser full-node is possible but hard (source)
    - Write the logic using portable language so it’s easy to move between platforms.
- Dynamic async (each party speaks only once (YOSO)) KeyGen.
- How to achieve the liveness of a p2p network without relying on a trusted third party.
    - We need at least one node with public IP or use STUN.
    - Use some public blockchain. But then who pays for the transaction fees? It may be solved similarly to how we did on stellot.com
- If we assume there exist more reliable and more trusted parties (aka. delegates, or representatives), we can do a few improvements:
    - Scalability, not everyone has to participate in MPC.
    - Easy dynamic DKG, since they are online, we don’t care about the number of rounds of DKG. We can re-share every time someone joins.
    - They can commit to running the nodes for the whole time of the elections, so we solve the liveness problem.

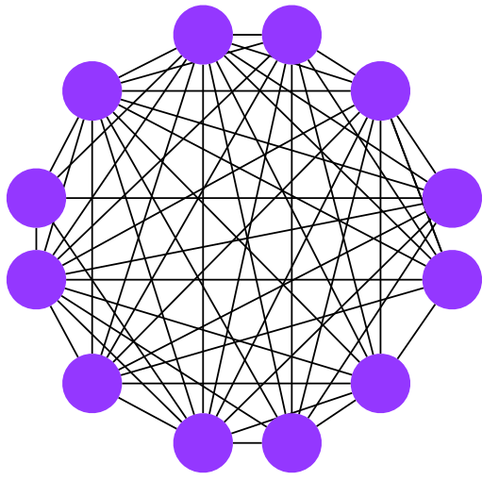
Delegated voter-to-voter model



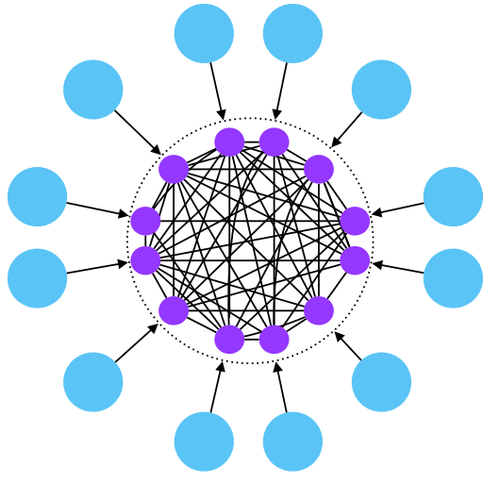
Trusted third party



Distributed third party



peer-to-peer



delegated peer-to-peer

Voter

Authority

Trust

Questions?

Stanislaw Baranski
stan@stan.bar
https://stan.bar

24.03.2023