E2E Verifiable
Decentralized Election

Sang Ho Lee, PH360, May 7 2018
Problem

- In current election, ballots are collected and analyzed by a centralized system
- This requires “Trust” in authority
Solution

Distributed Blockchain Network
Smart Contract

- Input: Data or Value
- (Contract—without middleman)
- Output: Finite Item from a Machine
Ethereum Virtual Machine (EVM)

- This is not Bitcoin
- Runtime environment for smart contracts in Ethereum
Immediate Solution

- End-to-End (E2E) Verifiability
- Individual Verifiability
- Universal Verifiability
“Self-tallying Elections and Perfect Ballot Secrecy” (Kiayias-Yung, 2002)

- A voter’s privacy can only be a collusion of all other voters
Quick Overview

- Each voter $P_i$ receives a secret value $x_i$
- All voters $P$ announces their voting key $g^x$ and $P(x)$ to prove their authenticity
Quick Overview (cont.)

$g^x_1, P(x_1),$ $g^x_2, P(x_2),$ $g^x_3, P(x_3),$ $\$
Quick Overview (cont.)

Reconstructed Key

\[ Y_i = \left( \prod_{j=1}^{i-1} g^{x_j} \right) / \left( \prod_{j=i+1}^{n} g^{x_j} \right) \]

\[ g^{y_3} = \left( g^{x_1} \times g^{x_2} \right) / \left( g^{x_4} \times g^{x_5} \right) \]
Quick Overview (cont.)

- All voters $P_i$ publish $g^{x_iy_i}g^{v_i}$ and show $v_i$ is either 1 or 0
- Using El Gamal encrypted vote, given $g^{xy}g^v$ and $g^v$,
- We can find $g^v$ is either $g^0$ or $g^1$ (no or yes)
Planned Goals

Ballots

- Show candidate options
- Allow only eligible voters

Pre-election ID verification

- External voter identification generates unique Ethereum address
- The address tied with the given ballot and smart contract
Planned Goals (cont.)

Voting

▸ Able to cast a vote
  Smart contract funded with Ether
▸ Votes are immutable to the chain

Election Result

▸ No more votes allowed after voting ends
▸ The tally is accessible and verifiable to the public
Biggest Problem

- Current End-to-End Verifiability only supports Boolean tally
- This is not what and election should look like
Additional “less-fancy” features

- **Self timer to end voting**
  
  Currently, the admin had to call out the vote

- **Not allowing multiple votes per account**
  
  Accounts could have casted multiple votes
Demo
Evaluation

Ballots

- Show candidate options ✓
- Allow only eligible voters ✓

Pre-election ID verification

- External voter identification generates unique Ethereum address ✗
- The address tied with the given ballot and smart contract ✓
Evaluation (cont.)

Voting

- Able to cast a vote ✓
  - Smart contract funded with Ether
  - Votes are immutable to the chain ✓
    + Not allowed to vote one than once

Election Result

- No more votes allowed after voting ends ✓
- The tally is accessible and verifiable to the public ✓
  + Self terminating election
Challenges

- First coding project
- Constantly updated, not backward compatible language & platform
- Unstable test-net (failed nodes)
- Lack of resources
Recognition & References

Special thanks to Professor Raja for instruction and guidelines

“Self-tallying Elections and Perfect Ballot Secrecy”
By A. Kiayias, Moti Yung


Tutorials from Dapp University (YT)

Ethereum Foundation

Numerous online forums and posts