Wariant 1: Jednoetapowe głosowanie

Rejestracja i głosowanie od razu

Głosujący - V
Serwer rejestracji - R
Serwer głosowania - B

Etap I - przygotowanie karty do głosowania

- V pozyskuje kartę do głosowania - BS_0 (dane publiczne)
- V wypełnia kartę - BS
- V szyfruje kluczem symetrycznym m (na razie posiadany tylko przez V) - $m(BS)$
- V zaślepia zaszyfrowaną kartę - $blinded(m(BS))$
- V podpisuje zaślepioną zaszyfrowaną kartę swoim kluczem prywatnym - $S^V_{priv}(blinded(m(BS)))$
- V szyfruje kluczem asymetrycznym serwera rejestracyjnego - $K^R_{pub}[S^V_{priv}(blinded(m(BS)))]$
- $K^R_{pub}[S^V_{priv}(blinded(m(BS)))]$ jest wysyłane do serwera rejestracyjnego R

Etap II - sprawdzenie i przyznanie prawa do głosowania

- R otrzymuje $K^R_{pub}[S^V_{priv}(blinded(m(BS)))]$ od V
- R odszyfrowuje wiadomość swoim prywatnym kluczem K^R_{priv} - $S^V_{priv}(blinded(m(BS)))$
- R identyfikuje V czytając jego podpis kluczem publicznym S^V_{pub} - $blinded(m(BS))$
    - R posiada listę wszystkich wyborców uprawnionych do głosowania, ale dla optymalizacji można dosłać o jakiego wyborcę chodzi np. w strukturze $K^R_{pub}[S^V_{priv}(blinded(m(BS), V))]$
- Jeśli w bazie nie ma tego wyborcy, zwracamy błąd; jeśli jest to postępujemy dalej
- R ślepo podpisuje $blinded(m(BS))$ swoim kluczem prywatnym do ślepego podpisu - $\sigma(blinded(m(BS)))$
- R podpisuje $\sigma(blinded(m(BS)))$ swoim kluczem prywatnym do podpisu - $S^R_{priv}[\sigma(blinded(m(BS)))]$
- $S^R_{priv}[\sigma(blinded(m(BS)))]$ jest wysyłane do wyborcy V

Etap III - przesłanie głosu do serwera głosowania

- V otrzymuje $S^R_{priv}[\sigma(blinded(m(BS)))]$ od R
- V sprawdza podpis R jego kluczem publicznym S^R_{pub} - $\sigma(blinded(m(BS)))$
- V odślepia wiadomość $\sigma(blinded(m(BS)))$ - $\sigma(m(BS))$
    - W ten sposób V ma $\sigma(m(BS))$, czyli podpisaną przez serwer rejestracji wypełnioną kartę do głosowania
- V podpisuje swoim kluczem prywatnym S^V_{priv} - $S^V_{priv}[\sigma(m(BS))]$

- V wysyła $S^V_{priv}[\sigma(m(BS))]$ do serwera głosowania B

Etap IV - odczytanie podpisów i publikacja zaszyfrowanej karty

- B otrzymuje $S^V_{priv}[\sigma(m(BS))]$
- B odczytuje podpisy V oraz R, potwierdzając tym wiarygodność karty
- B publikuje $m(BS)$
    - Przypomnienie: to jest dalej zaszyfrowana symetrycznie karta
- B może wysłać do V sygnał o opublikowaniu zaszyfrowanej karty

Etap V - weryfikacja opublikowanego głosu, jego ujawnienie i zliczenie

- V sprawdza czy jego głos ($m(BS)$) znajduje się na liście
- Jeśli nie, no to kłopot xd; jeśli jest no to fajnie, nasz głos jest na serwerze
- V wysyła klucz m do B, aby serwer odszyfrował głos
- B otrzymuje klucz symetryczny m, używa go do odszyfrowania głosu V i zlicza głos
- Po zakończeniu głosowania, publikowane są wyniki i opcjonalnie odszyfrowane karty (ponieważ są anonimowe)

Wariant 2: Dwuetapowe głosowanie

Składa się z dwóch etapów: 1. Rejestracja 2. Głosowanie

Głosujący - V
Serwer rejestracji - R
Serwer głosowania - B
Trust Centre - T

Etap I - głosujący zgłasza chęć do głosowania

- V generuje losowy token t
- V zaślepia token - $blinded(t)$
- V podpisuje zaślepiony token swoim kluczem prywatnym S^V_{priv} - $S^V_{priv}(blinded(t))$
- V szyfruje podpisany token kluczem publicznym centrum rejestracyjnego K^R_{pub} - $K^R_{pub}[S^V_{priv}(blinded(t))]$
- V wysyła to do serwera rejestracji R

Etap II - rejestracja głosującego

- R otrzymuje od V - $K^R_{pub}[S^V_{priv}(blinded(t))]$
- R odszyfrowuje swoim kluczem prywatnym K^R_{priv} - $S^V_{priv}(blinded(t))$
- R odczytuje podpis V za pomocą jego klucza publicznego S^V_{pub} - $blinded(t)$
- R podpisuje zaślepiony token - $\sigma_R(blinded(t))$
    - R może używać różnych kluczy do podpisywania zaślepionych tokenów dla każdego “obwodu wyborczego” c, ale można też wymyśleć inne rozwiązanie
- R zapisuje token (na wypadek gdyby głosujący próbował się ponownie zarejestrować, możemy wysłać ponownie ten sam token).
- R podpisuje ślepo podpisany token swoim kluczem prywatnym S^R_{priv} -

$S^{R}_{priv}(\sigma_{R}(blinded(t)))$

- R szyfruje wiadomość kluczem publicznym V K^{V}_{pub} - $K^{V}_{pub}[S^{R}_{priv}(\sigma_{R}(blinded(t)))]$
- R odsyła $K^{V}_{pub}[S^{R}_{priv}(\sigma_{R}(blinded(t)))]$ do V

Etap III - prośba o weryfikację do Trust Centre T

- Analogicznie jak w etapie I, V generuje inny token τ, zaślepia, podpisuje kluczem S^{V}_{priv} i szyfruje kluczem publicznym T K^{T}_{pub} - $K^{T}_{pub}[S^{V}_{priv}(blinded(\tau))]$
- V wysyła $K^{T}_{pub}[S^{V}_{priv}(blinded(\tau))]$ do Trust Centre T

Etap IV - potwierdzenie identyfikacji V przez Trust Centre

- Analogicznie jak w etapie II, T otrzymuje $K^{T}_{pub}[S^{V}_{priv}(blinded(\tau))]$, odszyfrowuje swoim kluczem prywatnym K^{T}_{priv}, potwierdza podpis V kluczem publicznym S^{V}_{pub} i podpisuje zaślepiony token - $\sigma_{T}(blinded(\tau))$
- T podpisuje ślepo podpisany token swoim kluczem prywatnym S^{T}_{priv} i szyfruje to kluczem publicznym V K^{V}_{pub} - $K^{V}_{pub}[S^{T}_{priv}(\sigma_{T}(blinded(\tau)))]$
- T wysyła do V $K^{V}_{pub}[S^{T}_{priv}(\sigma_{T}(blinded(\tau)))]$

Etap V - pozyskanie karty do głosowania

- V posiada zestaw tokenów $[t,\ \sigma_{R}(t),\ \tau,\ \sigma_{T}(\tau),\ c]$
- V generuje klucz symetryczny m
- V dokłada do zestawu tokenów identyfikator T oraz klucz m
    - Identyfikator T jest potrzebny, żeby zidentyfikować Trust Centre, z którego korzystał V tylko w przypadku, gdy istnieje więcej niż jedno Trust Centre
- V szyfruje zestaw kluczem publicznym K^{B}_{pub} - $K^{B}_{pub}[c,\ T,\ m,\ t,\ \sigma_{R}(t),\ \tau,\ \sigma_{T}(\tau)]$
- V przesyła prośbę o kartę do głosowania do serwera głosowania B $K^{B}_{pub}[c,\ T,\ m,\ t,\ \sigma_{R}(t),\ \tau,\ \sigma_{T}(\tau)]$

Etap VI - przesłanie zaszyfrowanej karty do głosowania

- B otrzymuje od V $K^{B}_{pub}[c,\ T,\ m,\ t,\ \sigma_{R}(t),\ \tau,\ \sigma_{T}(\tau)]$
- B odszyfrowuje informacje swoim kluczem prywatnym K^{B}_{priv} - $[c,\ T,\ m,\ t,\ \sigma_{R}(t),\ \tau,\ \sigma_{T}(\tau)]$
- B sprawdza podpisy σ_{R} i σ_{T} oraz sprawdza zgodność z jawnymi tokenami
    - Jeśli są niepoprawne to odrzucamy
    - Jeśli są poprawne to idziemy dalej
- B podpisuje pustą kartę do głosowania BS_{0} swoim kluczem prywatnym S^{B}_{priv} - $S^{B}_{priv}(BS_{0})$
- B szyfruje podpisaną kartę do głosowania symetrycznym kluczem m - $m[S^{B}_{priv}(BS_{0})]$
- B wysyła zaszyfrowaną podpisaną kartę $m[S^{B}_{priv}(BS_{0})]$ do V

Etap VII - oddanie głosu

- V otrzymuje $m[S^B_{priv}(BS_0)]$
- V odszyfrowuje kluczem symetrycznym m - $S^B_{priv}(BS_0)$
- V weryfikuje podpis kluczem publicznym S^B_{pub} - BS_0
- V wypełnia kartę do głosowania - BS
- V szyfruje zestaw $[c,\ T,\ m,\ t,\ \sigma_R(t),\ \tau,\ \sigma_T(\tau),\ BS]$ kluczem publicznym K^B_{pub} - $K^B_{pub}[c,\ T,\ m,\ t,\ \sigma_R(t),\ \tau,\ \sigma_T(\tau)]$
- V wysyła $K^B_{pub}[c,\ T,\ m,\ t,\ \sigma_R(t),\ \tau,\ \sigma_T(\tau)],\ BS$ do B

Etap VIII - przyjęcie i zliczenie głosu

- B przyjmuje $K^B_{pub}[c,\ T,\ m,\ t,\ \sigma_R(t),\ \tau,\ \sigma_T(\tau),\ BS]$
- B odszyfrowuje swoim kluczem prywatnym K^B_{priv} - $[c,\ T,\ m,\ t,\ \sigma_R(t),\ \tau,\ \sigma_T(\tau),\ BS]$
- B ponownie sprawdza podpisy σ_R i σ_T (tak jak w etapie VI)
- B zlicza głos z BS
- Po zakończeniu głosowania B publikuje wyniki
- Można opublikować listę wszystkich BS razem z podpisami σ_R i σ_T. Nie pozwala to na identyfikację wyborcy.

Do podpisów i szyfrowań asymetrycznych używamy RSA
Do szyfrowania symetrycznego używamy AES (lub coś innego)

Blind signature:
$blinded(m)$ - zaślepiona wiadomość
$\sigma(m)$ - podpisana wiadomość

m - wiadomość
$e,\ d,\ N$ - klucz publiczny, prywatny i modulus (z RSA) ($ed \equiv 1(mod\ N)$) od podpisującego
r - losowa liczba całkowita, taka że $r \in (2,\ N)$ i $gcd(r,\ N) = 1$

$$m' = blinded(m) = m \cdot r^e\ (mod\ N)$$
$$s' = \sigma(blinded(m)) = (blinded(m))^d = (m \cdot r^e)^d = m^d \cdot r^{ed} = m^d \cdot r$$
$$s = \sigma(m) = s' \cdot r^{-1} = m^d$$

Źródła:
https://ieeexplore.ieee.org/abstract/document/1174319
https://www.researchgate.net/publication/2895156_Electronic_Voting_Algorithmic_and_Implementation_Issues
https://en.wikipedia.org/wiki/Blind_signature
https://medium.com/rootstock-tech-blog/blind-signatures-af6338da6347