

Wykład 9. Niepokwitowalne protokoły głosowania

Wykładowca: Stefan Dziembowski

Skryba: Maria Fronczak, Edwin Vaina

Streszczenie. Na wykładzie była mowa o niepokwitowalnych protokołach głosowania, które mają na celu uniemożliwienie wpływania na głosy wyborców. Przedstawione zostały wymagania konieczne, by protokół głosowania mógł być niepokwitowalny, narzędzia do konstrukcji protokołu i wreszcie sam protokół, opisany w pracy [1].

1 Wstęp

Notatki na podstawie rozdziału 5. pracy [1].

Będziemy zajmować się *niepokwitowalnymi protokołami głosowania* (ang.: *receipt-free voting protocols*). Niepokwitowalne protokoły głosowania to protokoły głosowania, które nie tylko zapewniają poprawność wyników i tajność głosowania, ale w których dodatkowo głosujący nie jest w stanie udowodnić nikomu, jak zagłosował. Warunek taki zapobiega wpływaniu na wyborców poprzez kupowanie głosów bądź poprzez zmuszanie głosującego szantażem do głosowania według podanych wytycznych. Inaczej mówiąc w niepokwitowalnych protokołach głosowania wymagane są odporność na przekupstwo i na szantaż.

Przekupstwem nazywamy sytuację, która zachodzi, gdy ktoś kupuje głosy wyborców - to znaczy płaci głosującemu za głosowanie według podanych wytycznych. Głosujący dostaje pieniądze, gdy jest w stanie udowodnić, że zagłosował według tych wytycznych.

Szantaż jest scenariuszem silniejszym, w którym głosujący groźbą zmuszany jest do głosowania według wytycznych.

Różnica między szantażem a przekupstwem staje się widoczna, gdy istnieje możliwość poznania, jak naprawdę głosował głosujący. Nawet przy małym prawdopodobieństwie ujawnienia głosu szantażowany wyborca nie będzie ryzykował głosowania niezgodnie z wytycznymi szantażysty. Sytuacja, gdy możliwe jest poznanie głosu, jest

całkowicie nieakceptowalna dla wyborcy szantażowanego. Natomiast dla wyborcy przekupionego małe prawdopodobieństwo ujawnienia jego głosu może być akceptowalne, gdyż w razie głosowania niezgodnie z wytycznymi ryzykuje on co najwyżej, że nie dostanie pieniędzy (a i to zachodzi z niewielkim prawdopodobieństwem).

Dla ustalenia uwagi będziemy mówić o *szantażystce* - zatem poznanie, jak głosował wyborca nie może być możliwe.

Interesować nas będą wybory K -z- L kandydatów.

Model jest taki jak przed tygodniem (wyborcy V_1, \dots, V_M i organizatorzy A_1, \dots, A_N). Model ten trzeba jednak wzmocnić, jak zostało to niżej opisane.

2 Konieczne wymagania

2.1 Kanały niepodśluchiwalne

Będziemy wymagać, by między głosującymi a organizatorami istniały *fizycznie* bezpieczne kanały. Kanały takie nazywamy *kanałami niepodśluchiwalnymi*. Użycie kanałów niepodśluchiwalnych jest konieczne; nie wystarczy przesyłanie zaszyfrowanych wiadomości „zwykłymi” kanałami, bo szantażysta może zmusić wyborcę do ujawnienia mu klucza i odczytywać wszystkie przesyłane wiadomości. Przy użyciu kanałów niepodśluchiwalnych - w naszej sytuacji między wyborcą a organizatorami - głosujący nie jest w stanie udowodnić szantażystce, że coś przekazał.

Uzyskanie niepodśluchiwalnych kanałów wydaje się możliwe raczej przy głosowaniu z telefonów komórkowych niż poprzez internet, przy założeniu, że szantażysta nie ma dostępu do tego, co jest wysyłane z telefonów komórkowych.

2.2 Nieprzekupność organizatorów

Będziemy też wymagać żeby organizatorzy nie kolaborowali z szantażystą. Jest to konieczne: głosujący, który chce oszukać szantażystę co do swego głosu, musi kłamać co do przebiegu komunikacji z co najmniej jednym organizatorem. Jeśli jakikolwiek organizator współpracuje z szantażystą, istnieje niezerowe prawdopodobieństwo, że szantażysta złapie głosującego na kłamstwie (jeśli głosujący będzie kłamał co do przebiegu komunikacji z tym właśnie organizatorem); a jak zostało wcześniej powiedziane, sytuacja taka jest nieakceptowalna.

3 Komunikacja

1. Zakładamy, że istnieje PKI i że każdy wyborca zna swój klucz prywatny (ale może go ujawnić komu chce, w szczególności szantażyście). Chcemy uniknąć sytuacji, gdy szantażysta wybiera za głosującego klucz prywatny i publiczny, tak że głosujący nie zna nawet swojego klucza prywatnego. W dalszej części pokazany będzie więc protokół zapewniający, że głosujący zna swój klucz prywatny.
2. Tablica ogłoszeń, zdefiniowana na poprzednim wykładzie: każdy może pisać w wyznaczonej dla siebie części, wszyscy mogą czytać z tablicy, ale nikt nie może z niej niczego usuwać.
3. Niepodsluchiwalne kanały, jakie zostały wcześniej opisane (nie muszą być autentykowane).

4 Narzędzia

4.1 Szyfr homomorficzny

Pojęcie *szyfru homomorficznego* zostało wprowadzone na wcześniejszym wykładzie. Szyfr $E_Z: V \times R \rightarrow E$ (gdzie Z , zazwyczaj pomijane przy zapisie, jest kluczem publicznym, V jest zbiorem wiadomości, R jest losowym wejściem, zaś E zbiorem kryptogramów) jest homomorficzny, gdy dla dowolnych wiadomości v_1 i v_2 oraz losowości α_1 i α_2 zachodzi

$$E(v_1, \alpha_1) + E(v_2, \alpha_2) = E(v_1 + v_2, \alpha_1 + \alpha_2)$$

Wprowadzamy dodatkowo pojęcie *q-odwracalności*.

Definicja 1 *Szyfr jest q-odwracalny dla $q \in \mathbb{Z}$, gdy dla każdego szyfrogramu e , wiadomości v i losowości α umiemy efektywnie obliczyć qe - tzn. mając q i qe możemy obliczyć v i α takie, że $E_z(v, \alpha) = qe$ nie znając klucza prywatnego.*

Dla $q = 1$ jest to po prostu odszyfrowanie, rozważa się więc raczej duże q .

4.2 Σ -dowody

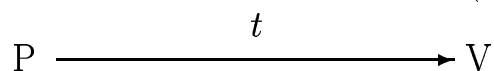
4.2.1 Pojęcie Σ -dowodu

Σ -dowody są to dowody wiedzy z wiedzą zerową specjalnej postaci.

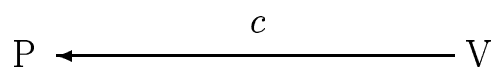
Prover P dowodzi, że zna ξ spełniające predykat $Q(\xi)$; na początku P zna Q i ξ takie, że zachodzi $Q(\xi)$, zaś weryfikator V zna Q .

Dowód przebiega w trzech etapach:

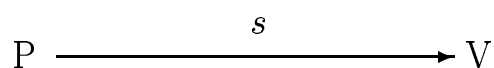
1. P wyznacza t na podstawie ξ (i pewnej losowości) oraz wysyła t do V:



2. V losuje c i wysyła c do P:



3. P wyznacza s (na podstawie wartości z punktów 1. i 2.) oraz wysyła s do V:



Na podstawie (t, c, s) V akceptuje lub nie.

W praktyce często stosowane są nieinteraktywne Σ -dowody, w których losowe wyzwanie c zastępowane jest wartością funkcji haszującej od pierwszej wiadomości wysłanej przez P (t).

Wobec Σ -dowodu wymagamy spełnienia następujących warunków:

1. pełność:

Jeśli P i V są uczciwi i P zna ξ , to V akceptuje.

2. poprawność:

Jeśli P potrafi odpowiedzieć na 2 różne wyzwania c , to możemy obliczyć ξ .

(Tzn. istnieje ekstraktor wiedzy obliczający ξ : mając dane dwie konwersacje (t, c_1, s_1) i (t, c_2, s_2) takie, że $c_1 \neq c_2$ umie wydajnie obliczyć ξ takie, że $Q(\xi)$.)

3. „specjalna” wiedza zerowa:

W przypadku Σ -dowodów mamy do czynienia z nieco inną „wiedzą zerową” niż dotychczas. Wymagamy mianowicie, by istniał symulator S , który dla każdego c wyprodukuje t i s takie, że (t, c, s) będzie mieć taki sam rozkład, jak przy dowodzie, w którym P i V są uczciwi (w szczególności w którym V wybiera c zupełnie losowo).

Σ -dowód przedstawiamy formalnie jako czwórkę (π_t, u, π_s, ϕ) , gdzie:

- π_t jest algorytmem produkującym t (na podstawie ξ i losowości r)
- u jest takie, że losowe $c \in Z_u$

- π_s jest algorytmem produkującym s (na podstawie c, ξ i r)
- ϕ jest algorytmem stwierdzającym, czy V akceptuje (t, c, s)

4.2.2 Redukcje przestrzeni wyzwań

Dla dowodu Σ znajomości świadka predykatu Q , gdzie $\Sigma = (\pi_t, u, \pi_s, \phi)$, przestrzenią wyzwań jest Z_u takie, że wyzwanie $c \in Z_u$.

Dla dowolnego dowodu $\Sigma' = (\pi_t, u', \pi_s, \phi)$, gdzie $u' < u$ (przy czym u' jest duże, czyli $\frac{1}{u'}$ jest zaniedbywalne) Σ' także jest Σ -dowodem znajomości świadka Q - spełnienie przez Σ' wszystkich warunków Σ -dowodu wynika ze spełnienia tych warunków przez dowód Σ .

4.2.3 AND-kombinacje

Mając dany Σ -dowód dla Q ze świadkiem ξ i Σ -dowód dla Q' ze świadkiem ξ' możemy uzyskać dowód dla $Q \wedge Q'$.

Jeśli u określa przestrzeń wyzwań w dowodzie dla Q , zaś u' określa przestrzeń wyzwań w dowodzie dla Q' , przyjmujemy $u^* = \min(u, u')$. Dowody dla Q i Q' biegną równolegle aż do wysłania wyzwania. Stosujemy wówczas redukcję przestrzeni wyzwań w obu dowodach do Z_{u^*} ; w obu dowodach wysyłane jest to samo $c \in Z_{u^*}$. V akceptuje, jeśli $\phi(t, c, s) \wedge \phi(t', c, s')$.

4.2.4 OR-kombinacje

Mając dane Σ -dowody dla Q i Q' jak dla AND-kombinacji, możemy skonstruować dowód dla $Q \vee Q'$, nie mówiąc, który z dowodów - ten dla Q czy ten dla Q' - znamy. Bez straty ogólności możemy założyć, że P zna dowód dla Q : zatem na początku P zna Q, Q' i ξ takie, że $Q(\xi)$, zaś V zna Q i Q' .

1. P losuje r i r' - losowe wejścia dla odpowiednich algorytmów i $c' \in Z_{u^*}$, gdzie u^* zdefiniowane jak wcześniej.

Następnie podstawia $(t', c', s') = S'(c', r')$, gdzie S' jest symulatorem opisanym wcześniej; oblicza też $t = \pi_t(\xi, r)$.

P wysyła do V t i t' :

$$P \xrightarrow{t, t'} V$$

V nie wie, czy to t zostało wyprodukowane uczciwie, a t' wygenerowane przez symulator, czy na odwrót.

2. V losuje $c^* \in Z_{u^*}$ i wysyła do P :

$$P \xleftarrow{c^*} V$$

3. P przyjmuje $c = (c^* - c') \bmod u^*$ oraz $s = \pi_s(\xi, c, r)$ (s jest tym, co P by uczciwie odpowiedział) oraz wysyła do V c, c', s i s' :

$$P \xrightarrow{c, c', s, s'} V$$

V sprawdza, czy $c = c' + c^*$ i czy $\phi(t, c, s)$ lub $\phi(t', c', s')$ - jeśli tak, akceptuje.

4.3 Schemat identyfikacji

Przedstawiony został schemat identyfikacji Schnorra.

Identyfikacja następuje przez udowodnienie znajomości logarytmu dyskretnego jakiejś liczby. W grupie G z trudnym logarytmem dyskretnym brane jest $Z = g^x$, Z jest kluczem publicznym, zaś x kluczem prywatnym (g jest oczywiście generatorem grupy G). Oznaczamy $q = |G|$.

By udowodnić swą tożsamość, P musi udowodnić, że zna x takie, że $g^x = Z$:

1. P losuje $r \in Z_q$, oblicza $t = g^r$, które wysyła do V .

2. V wysyła do P losowe $c \in Z_q$.

3. P wysyła do V $s = cx + r$.

V sprawdza, czy $g^s = Z^c \cdot t$, jeśli tak, akceptuje.

W ten sposób dostajemy schemat identyfikacji, który jest Σ -dowodem.

4.4 Dowód z desygnowanym weryfikatorem

Dążymy do tego, by V , kiedy P coś mu udowodni, nie mógł potem powtórzyć tego dowodu komuś innemu.

P ma dowieść znajomości ξ takiego że zachodzi $Q(\xi)$. Aby V nie mógł powtórzyć nikomu dowodu, P dowodzi, że zna ξ takie, że zachodzi $Q(\xi)$ lub że zna x takie, że $Z = g^x$ (gdzie Z jest kluczem publicznym weryfikatora, a x jego kluczem prywatnym). V wie, że P nie zna x , które jest kluczem prywatnym V . Zatem V wie, że P dowodzi, że zna ξ .

Gdyby jednak V chciał przedstawić komuś przeprowadzony dowód jako dowód znajomości ξ , nie może tego zrobić, ponieważ V zna swój własny klucz prywatny x . Dlatego warunek, by V znał swój klucz prywatny (a w naszym wypadku - by głoszący znał swój klucz prywatny) jest konieczny.

4.4.1 Zapewnienie, że głoszący zna swój klucz prywatny

Należy więc zapewnić, by głoszący znał swój klucz prywatny. Robimy to następująco:

1. Każemy głoszącemu podzielić swój głos schematem Shamira pomiędzy organizatorów A_1, \dots, A_n . (Oczywiście nie wiemy w tym miejscu, że to rzeczywiście głoszący, a nie szantażysta, podzielił klucz prywatny głoszącego.)
2. Organizatorzy wysyłają swoje udziały z powrotem do głoszącego (zakładamy, że wszystkie udziały dojdą do głoszącego).

4.5 Szyfr ElGamala

Przedstawiony tu szyfr będzie nieco inny niż na wcześniejszym wykładzie.

Bierzemy grupę G , $|G| = q$. Szyfr $E_Z: Z_q \times Z_q \rightarrow G \times G$ działa następująco:

$$E_Z(v, \alpha) = (g^\alpha, \gamma^v \cdot Z^\alpha)$$

gdzie Z jest kluczem publicznym, v jest wiadomością, α jest losowe, zaś g i γ są generatorami grupy G . Dodatkowo $Z = g^z$, z jest kluczem prywatnym.

Odszyrowanie przebiega w następujący sposób: skoro znamy z , to także $(g^\alpha)^z = (Z)^\alpha$. Dostajemy więc γ^v , które trzeba jeszcze zlogarytmować, by otrzymać wiadomość v (co dla małego v nie jest problemem).

Podany szyfr jest q -odwracalny ($q = |G|$). Dla dowolnego elementu e grupy G $e^q = 1$. Dla $(v, \alpha) = (0, 0)$ mamy $E_z(0, 0) = (1, 1)$, zatem dla bezpieczeństwa q powinno być duże.

Alternatywnie stosować można szyfr Paillera, który na wykładzie nie został omówiony.

4.6 Reenkrypcja

Definicja 2 Reenkrypcja e' szyfrogramu e , gdzie e jest szyfrogramem pewnej wiadomości m z losowością r : $e = E(m, r)$ nazywamy szyfrogram tej samej wiadomości m z inną losowością r' : $e' = E(m, r')$.

By pokazać, że $e' = E(m', r')$ jest reenkrypcją $e = E(m, r)$, korzystamy z tego, że E jest szyfrem homomorficznym:

$$E(m, r) - E(m', r') = E(m - m', r - r') = E(0, r - r')$$

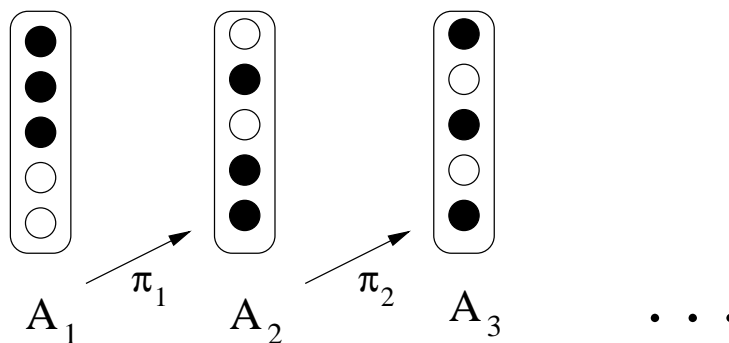
jeśli $m = m'$. (Zapisujemy wówczas $e' = R(e, r - r') = e + E(0, r - r')$.)

Należy więc pokazać, że $e - e'$ jest szyfrogramem dla wiadomości 0. Dowód tego częściowo został pokazany na ćwiczeniach; zamieszczony też jest w dodatku na końcu tej notatki.

5 Protokół wyboru K-z-L

5.1 Zarys protokołu

Ponizej przedstawiony jest właściwy protokół wyboru K spośród L kandydatów. Kandydaci są ponumerowani od 1 do L. W „zerowej” turze głosujący dostaje listę kandydatów i głosuje na pierwszych K kandydatów (to znaczy „zerowa” karta do głosowania ma ciemną kropkę - wartość „1” oznaczającą oddanie głosu na danego kandydata na K pierwszych miejscach i jasną kropkę - „0” na pozostałych miejscach.):



W kolejnych krokach permutujemy listę z pomocą kolejnych organizatorów A_1, A_2 itd. W pierwszym z takich kroków głosujący V wysyła od organizatora A_1 permutację π_1 , zgodnie z którą A_1 ma spermutować „zerową” kartę głosowania V .

Otrzymana po takiej permutacji karta jest następnie permutowana przez A_2 z użyciem permutacji π_2 , którą A_2 otrzymał od V itd. A_1 wie oczywiście, jaka jest wartość pierwszej karty do głosowania V . Jednak już aby poznać wartość drugiej karty do głosowania V A_1 i A_2 muszą kolaborować. Iterując algorytm dostatecznie wiele razy trafimy wreszcie na uczciwego organizatora, a jeden uczciwy organizator wystarczy, by wartość karty do głosowania pozostała nieznaną.

5.2 Dokładny opis protokołu

Oznaczamy przez t liczbę uczciwych organizatorów (to znaczy poprawny wynik wyborów otrzymujemy, gdy t organizatorów jest uczciwych, zaś dla zachowania tajności

co najwyżej $t - 1$ organizatorów może współpracować).

Karty (czy dokładniej wartości na nich) szyfrujemy szyfrem E , wartości zaszyfrowane publikowane są na tablicy ogłoszeń.

Zaczynamy więc od karty $(\underbrace{E(1, 0), \dots, E(1, 0)}_K, \underbrace{E(0, 0), \dots, E(0, 0)}_{L-K})$, którą zapisujemy jako (e_1^0, \dots, e_L^0) . W kolejnym kroku k dokonywane jest przekształcenie karty do głosowania numer $k - 1$ w kartę do głosowania numer k :

$$\begin{bmatrix} e_1^{(k-1)} \\ \cdot \\ \cdot \\ e_L^{(k-1)} \end{bmatrix} \xrightarrow{A_k} \begin{bmatrix} e_1^{(k)} \\ \cdot \\ \cdot \\ e_L^{(k)} \end{bmatrix}$$

gdzie e_i^k jest reenkrypcją $e_{\pi_k(i)}^{(k-1)}$.

W każdym kroku k potrzebne są dwa dowody:

1. dowód publiczny, że A_k rzeczywiście spermutował jakąś kartę (czyli że nowa lista jest poprawną kartą wyboru K-z-L).
2. dowód dla V , że A_k rzeczywiście spermutował poprzednią kartę V zgodnie z permutacją π_k , którą otrzymał od V .

5.2.1 Dowód poprawności karty

Organizator A_k musi dowieść, że jeśli poprzednia karta wyborcza (numer $k - 1$) była poprawna, to karta wyprodukowana przez niego, k -ta, też jest poprawna. Dokładniej dowieść musi dwóch rzeczy:

1. że wartości na wyprodukowanej karcie są reenkrypcjami wartości z poprzedniej karty (numer $k - 1$).
2. że suma wartości na wyprodukowanej karcie jest reenkrypcją sumy wartości z poprzedniej karty.

Dowód, że wartości na karcie numer k są reenkrypcjami wartości z karty $k - 1$. A_k musi wykazać, że $\forall i \in 1 \dots L$ zachodzi, że $e_i^{(k)}$ jest reenkrypcją któregoś z $e_1^{(k-1)}, \dots, e_L^{(k-1)}$ (oczywiście nie może pokazać którego, by nie ujawnić permutacji).

Definiowany jest predykat Q : $Q_{i,j}(\xi) \iff e_j^k = R(e_j^{(k-1)}, \xi)$,
gdzie $R(e_j^{(k-1)}, \xi) = e_j^{(k-1)} + E(\xi, 0)$. Organizator musi pokazać, że zachodzi

$$\Theta(\xi_1, \dots, \xi_L) = (Q_{11}(\xi_1) \vee \dots \vee Q_{1L}(\xi_1)) \wedge \dots \wedge (Q_{L1}(\xi_L) \vee \dots \vee Q_{LL}(\xi_L))$$

co czyni korzystając z opisanych narzędzi (dowód reenkrypcji, AND- i OR-kombinacje).

Dowód, że suma wartości na karcie numer k jest reenkrypcją sumy wartości z karty $k - 1$. Organizator A_k powinien też pokazać, że $e_\Sigma^{(k)} = \text{sum}_{i=1}^L e_i^{(k)}$ jest reenkrypcją $e_\Sigma^{(k-1)} = \text{sum}_{i=1}^L e_i^{(k-1)}$. Robi to korzystając z homomorfizmu E : jeśli losowości w enkrypcji poszczególnych elementów sumy $e_\Sigma^{(k-1)}$ to ξ_1, \dots, ξ_L , to pokazuje, że $e_\Sigma^{(k)}$ jest reenkrypcją $e_\Sigma^{(k-1)}$ z losowością $\xi_1 + \dots + \xi_L$.

5.2.2 Dowód zgodności z permutacją

Zauważmy, że głos wyborcy określony jest jednoznacznie przez „zerową” kartę do głosowania i pewną permutację π . Głosujący dzieli π na t losowych permutacji π_1, \dots, π_t , $\pi = \pi_1 \pi_2 \dots \pi_t$ i używa organizatorów do spermutowania jego kart do głosowania. Jeśli wśród t organizatorów co najmniej jeden jest uczciwy, permutacja π pozostanie nieznaną przeciwnikowi. (W wypadku, gdy wyborca stwierdzi, że któryś z organizatorów jest nieuczciwy, zwraca się z tą samą permutacją do innego organizatora. W takim wypadku obwieszcza też publicznie, że ten organizator jest nieuczciwy i że ma być pominięty przy liczeniu głosów; głosujący ma prawo do co najwyżej $t - 1$ takich obwieszczeń.)

Z drugiej strony, wyborca nie może mieć dowodu na to, że rzeczywiście użył permutacji π jeśli protokół ma być niepokwitowalny; dowód zgodności z permutacją powinien być przekonujący tylko dla głosującego i dla nikogo innego. Wobec tego organizator dowodzący zgodności z permutacją używa dowodu z desygnowanym weryfikatorem, pokazując, że albo karta, którą wyprodukował, jest reenkrypcją podanej permutacji poprzedniej karty, albo że zna klucz prywatny głosującego.

Dla głosującego jest to oczywiście dowód zgodności z permutacją. Skoro jednak zapewniliśmy, że głosujący zna swój klucz prywatny, nie może on udowodnić nikomu innemu, że jest to dowód zgodności z daną permutacją.

5.3 Obliczanie wyników wyborów

Na końcu liczone są po współrzędnych sumy głosów (czyli liczby głosów na poszczególnych kandydatów), korzystając się z tego, że używany szyfr był homomorficzny. Następnie sumy te są rozszyfrowywane. Niestety, złożoność odszyfrowania (wiążąca się z obliczeniem logarytmu dyskretnego) jest liniowa względem liczby głosujących (ewentualnie proporcjonalna do pierwiastka z liczby wyborców).

Poza tym algorytm ten, inaczej niż algorytm pokazany na poprzednim wykładzie, wymaga by każdy wyborca kontaktował się z t organizatorami.

Oczywiście obliczenie wyników wyborów może ujawnić, że głosujący głosował niezgodnie ze wskazówkami szantażysty, gdy np. szantażysta żądał oddania głosu na kandydata nr. 1 (i dowolnych $K - 1$ innych), a kandydat nr. 1 w ogóle nie dostał głosów. Jednak prawdopodobieństwo, że podobna sytuacja wystąpi, jest pomijalne.

Literatura

[1] Hirt M. *Multi-Party Computation: Efficiency Protocols, General Adversaries, and Voting*. Hartung-Gorre Verlag Konstanz, 2001.

A Dowód reenkrypcji

Opisany niżej dowód jest bardziej ogólny niż wykazanie po prostu, że $e - e'$ jest szyfrogramem wiadomości 0: pokazywane jest, że dla każdego kryptogramu e i wiadomości (głosu) v P zna ξ takie, że $e = E(v, \xi)$.

P zna e, v, ξ takie, że $e = E(v, \xi)$, V zna e i v .

1. P losuje α i podstawia $e' = E(0, \alpha)$. Wysyła e' do V.
2. V wysyła do P losowe $c \in Z_q$, przy czym q jest pierwsza.
3. P wysyła do V β takie, że $\beta = c \cdot \xi + \alpha$.
V sprawdza, czy $E(c \cdot v, \beta) = c \cdot e + e'$, jeśli tak, akceptuje.

A.1 Pełność

Jeśli P i V są uczciwi i , to V zaakceptuje:

$$E(c \cdot v, \beta) = E(c \cdot v, c \cdot \xi + \alpha) = E(c \cdot v, c \cdot \xi) + E(0, \alpha) = c \cdot E(v, \xi) + E(0, \alpha) = c \cdot e + e'$$

gdzie w kolejnych przejściach wykorzystywany był fakt, że E jest homomorficzne.

A.2 Poprawność

Jeśli P umie odpowiedzieć na 2 różne wyzwania c , to potrafi wydajnie obliczyć ξ .

Niech (e', c, β) i (e', c', β') , gdzie $c \neq c'$ będą transkryptami dwóch dowodów, w których V akceptuje.

Jeśli V akceptuje (e', c, β) i (e', c', β') to znaczy, że

$$E(cv, \beta) = ce + e' \text{ oraz } E(c'v, \beta') = c'e + e'$$

skąd

$$E((c - c')v, \beta - \beta') = (c - c')e$$

Możemy przyjąć bez straty ogólności, że $c > c'$, zatem $0 < c - c' < u$ i $\gcd(c - c', q) = 1$, bo q jest pierwsza.

Stosując algorytm Euklidesa możemy więc znaleźć a i b takie, że

$$a(c - c') + bq = 1$$

Z q -odwracalności zastosowanego szyfru możemy obliczyć (v_q, α_q) takie, że $e \cdot q = E(v_q, \alpha_q)$, a także

$$\begin{aligned} e &= (a(c - c') + bq)e = a(c - c')e + bqe = aE((c - c')v, \beta - \beta') + bE(v_q, \alpha_q) \\ &= E(a(c - c')v + bv_q, a(\beta - \beta') + b\alpha_q) = E(a(c - c')v + bq v, a(\beta - \beta') + b\alpha_q) \\ &= E(v, a(\beta - \beta') + b\alpha_q) \end{aligned}$$

gdzie skorzystaliśmy z tego, że $qv = v_q$, ponieważ E jest homomorficzne.

Zatem jeśli P umie odpowiedzieć na wyzwania c i c' , $c \neq c'$, to wie, że e jest szyfrogramem wiadomości v z losowością $\xi = a(\beta - \beta') + b\alpha_q$.

A.3 Specjalna wiedza zerowa

Symulator działa następująco: dla danego $c \in Z_u$, β jest wybierane losowo i przyjmowane jest $e' = E(cv, \beta) - ce$.