Протокол тайного голосования «Все против одного»

Основная идея состоит в том, что избиратели сначала составляют партию голосов, проставляя каждому кандидату случайное число (возможно отрицательное) голосов, затем, – такую партию, чтобы, при сложении с первой, получился только один голос за одного кандидата, которого они выбрали. Далее, каждую партию голосов разделяют еще на партии голосов, равную количеству избирателей, и подписывают уникальным кодом, отправляют одному из избирателей. Затем, каждый избиратель сдает чужие партии голосов счетчику.

Алгоритм

Условные обозначения:

V – регистратор (validator), C – счетчик (counter), P_i – избиратель, n – кол-во избирателей, k – кол-во кандидатов, IP_i – IP адрес избирателя

$A_1 \ldots A_i \ldots A_n$ – голоса избирателей, где A_i – вектор из k чисел, из которых только одна единица, все остальные нули (единица, которая стоит с порядковым номером N, означает, что избиратель проголосовал, за кандидата с порядковым номером N)

$A_i = B_i + C_i$, где B_i - первая часть голоса, C_i – вторая часть голоса. B_i и C_i - тоже вектора из k чисел, каждое число может принимать значение от (-4*n) до (4*n)

B_{ij} – j-ая часть первой части голоса избирателя P_i. $B_i = \sum_{j=1}^{n} B_{ij}$. C_i - аналогично $C_i = \sum_{j=1}^{n} C_{ij}$

$A_i = B_i + C_i = \sum_{j=1}^{n} B_{ij} + \sum_{j=1}^{n} C_{ij}$ Пример: Пусть в выборах участвует 4 кандидата и 2 избирателя и P_1 отдает свой голос за 3-его кандидата, тогда $A_1 = (0,0,1,0)$, $B_1 = (3,-3,2,0)$ (заполняется случайно), $C_1 = (-3,3,-1,0)$, $B_1 = (2, -1, 1, 0) + (1,-2,1,0)$, $C_1 = (-2, 1, 0, 0) + (-1,2,-1,0)$

Итого: $A_1 = (2, -1, 1, 0) + (1,-2,1,0) + (-2, 1, 0, 0) + (-1,2,-1,0) = (0,0,1,0)$

1. V составляет список избирателей и кандидатов. Генерирует закрытый и открытый ключ ($V_{private}$ и V_{public}) Публикует список кандидатов и V_{public}. Дожидается подключения всех P_i и C

2. P_i шифрует свои личные данные ($name_i$) ключом V_{public} и отправляет их V. Генерирует ($P_{iprivate}$ и $P_{ipublic}$), публикует $P_{ipublic}$

3. V расшифровывает ($name_i$)* ключом $V_{private}$, если ($name_i$) был в списке избирателей, сохраняет IP_i

4. C генерирует закрытый и открытый ключ ($C_{private}$ и C_{public}) подключается к V

5. Когда к V подключатся все P_i и C, V шифрует все (IP_i) ключом C_{public} и передает C.

   Генерирует n таких векторов D_i, что $\sum_{i=1}^{n} D_i = 0$, и подписывает их уникальным шифром

   > В систему вводятся D_i - голоса «мертвых душ», чтобы, если все (P_i, $i \neq l$) сговорились против одного P_l, они не смогли узнать его голос

   Каждому P_i передает все (IP_i) и D_i, зашифрованные ключом $P_{ipublic}$

6. P_i получает и расшифровывает список из (IP_i)*, делает свой выбор A_i и составляет случайным образом B_i, а C_i рассчитывает ($C_i = A_i - B_i$).
   B_i и C_i разбивает на части B_{i1}+…+B_{ij}+…B_{in} и C_{i1}+…+C_{ij}+…C_{in}. Каждую часть B_{ij} и C_{ij} подписывает уникальным кодом, шифрует сначала ключом C_{public}, затем ключом $P_{jpublic}$ наборы: (УК+B_{ij}), (УК+C_{ij}) и передает зашифрованное каждому P_j, используя IP_j

7. Каждый P_j дожидается получения n пакетов из ((УК+B_{ij})*)*, ((УК+C_{ij})*)*, расшифровывает полученное ключом $P_{jprivate}$ и отправляет счетчику ((УК+B_{1j})*+…+ (УК+B_{nj})*)+((УК+C_{1j})*+…+ (УК+C_{nj})*)+D_j*

8. C получает n пакетов (УК+B_{ij})*, (УК+C_{ij})* и D_j*, проверяет, что они пришли от пользователя из IP_i, расшифровывает их ключом $C_{private,}$ публикует (УК+B_{ij}), (УК+C_{ij}) и D_j, подсчитывает результат голосования $A = \sum_{i=1}^{n}\sum_{j=1}^{n} B_{ij} + \sum_{i=1}^{n}\sum_{j=1}^{n} C_{ij} + \sum_{i=1}^{n} D_i$ Если сумма чисел вектора A не равна n, то результат считается фальсифицированным. Публикует результат A

Критический анализ

1. V: IP_i <-> $name_i$
   > V знает связь, имя каждого избирателя и его адрес. Больше ему ничего не известно

2. P_i: IP_i <-> (B_{ij} и C_{ij})* или (C_{ij} и B_{ij})* или (B_{ij} и B_{ij})* или (C_{ij} и C_{ij})*
   > Каждый отдельный P_i знает, с какого адреса ему пришел пакет с кусочком голоса, но он не знает имени человека с известным адресом, и содержимое пакета тоже не известно, т.к. первоначально он зашифрован ключом $C_{public.}$
   > Чтобы V и C не могли перехватить пакет от одного избирателя к другому, он шифруется ключом $P_{jpublic}$ (получателя)

3. P_2…P_n: IP_i <-> (B_{i2}…B_{in} и C_{i2}…C_{in})*
   > Если все избиратели, кроме одного (P_1) сговорятся против P_1, то они смогут собрать все кусочки, кроме тех, которые P_1 отправляет счетчику сам.
   > Сговорившиеся избиратели смогли бы узнать голос P_1, но все собранные кусочки зашифрованы

4. C: IP_i | (B_i и C_i) = A_i
   > C знает, с какого адреса ему пришел набор пакетов голосов, из которых 2 пакета принадлежат отправителю, но не знает какие именно.
   > C не может подменить голоса, так как каждый кусочек голоса подписан и его надо опубликовать

5. C, P_2…P_n: IP_i <-> A_i

   Если C сговорится со всеми избирателями против P_1, то они смогут узнать голос P_1 и его адрес, но не узнают имя

6. V, C, P_2…P_n: $name_i$ <-> IP_i <-> A_i

   Если сговорятся V, C и все избиратели против P_1, то они смогут узнать голос, адрес и имя P_1. Из этого вывода я решил дать протоколу название «Все против одного».